TO 7 TO7 MO5

INITIATION A LOGO

DORIS AVRAM - MICHÈLE WEIDENFELD

cedic
nathan

TO TEK International

INITIATION A LOGO

Doris Avram et Michèle Weidenfeld
et l'équipe de S.O.L.I.
Illustration de couverture : Nadine Monnier

TO TEK International
cedic nathan

Dans la même collection

Initiation au BASIC TO7 — *Christine et François-Marie Blondel.*

Le BASIC D.O.S. du TO7 — *Christine et François-Marie Blondel.*

Un ordinateur à la maison — *Jean Delcourt.*

Un ordinateur en fête — *Serge Pouts Lajus.*

Un ordinateur et des jeux — *Jean-Pascal Duclos.*

LOGO, Manuel de référence — *Doris Avram — Tristan Savatier — Michèle Weidenfeld et l'équipe de S.O.L.I.*

Manuel Technique du TO7 et TO7-70 — *Michel Oury*

Guide du MO5 — *André Deledicq*

Faites vos jeux en assembleur sur TO7 et TO7-70 — *Michel Oury*

Manuel de l'assembleur du 6809 et du TO7 et TO7-70 — *Michel Weissgerber*

Initiation au FORTH — S.E.F.I.

Manuel technique du MO5 — *Michel Oury*

Ce volume porte la référence
ISBN 2-7124-0532-3

Toute reproduction, même partielle, de cet ouvrage est interdite. Une copie ou reproduction par quelque procédé que ce soit, photographie, photocopie, microfilm, bande magnétique, disque ou autre, constitue une contrefaçon passible des peines prévues par la loi du 11 mars 1957 sur la protection des droits d'auteur.

© *CEDIC 1984*

CEDIC, 6-10 boulevard Jourdan, 75014 - PARIS

Ce manuel a été rédigé par :
Doris Avram et Michèle Weidenfeld
sous la direction de :
Gérard Dahan
avec la participation de :
Sylvain Angeli et Marie-Paule Deprund
avec l'aide de :
Anahid Derian, Eveline Rosenfeld et toute l'équipe de S.O.L.I. — Paris.

L'implantation du langage Logo a été réalisée sous la direction de :
Gérard Dahan
par :
Sylvain Angeli, avec la collaboration de Marie-Paule Deprund, Jean-Baptiste Touchard, Christian Jullien et l'équipe de S.O.L.I. — Paris.

Nous tenons à remercier le professeur Seymour Papert pour les suggestions apportées à la rédaction de ce manuel.
Nous tenons à remercier l'équipe de Thomson-Simiv et plus particulièrement Bernard Dion et Tristan Savatier pour leur coopération et leurs précieux conseils.

Le logiciel contenu dans la cartouche LOGO ainsi que cet ouvrage sont protégés par un copyright de S.O.L.I. (Paris) et L.C.S.I. (Montréal).
La garantie offerte par S.O.L.I. se limite à la garantie exprimée dans les termes des articles 1641 et suivants du code civil français.
S.O.L.I. a fait tous les efforts pour assurer la validité la meilleure du logiciel informatique ainsi que de cet ouvrage à la date de la publication. S.O.L.I. ne saurait cependant être tenu pour responsable des conséquences liées à toute erreur ou omission dans ce livre.
Les caractéristiques de ce produit sont susceptibles d'amélioration, de mise à jour, et de modifications sans préavis. S.O.L.I. ne saurait être tenu pour responsable des conséquences éventuelles directes ou indirectes que pourraient entraîner de telles améliorations ou un usage non conforme à sa destination du logiciel LOGO.

© *S.O.L.I. (Paris) 1984*
Systèmes d'Ordinateur Logo International
33, rue de Poissy
Paris 75005, France

© *L.C.S.I. (Montréal) 1984*
Logo Computer Systems Inc.
9960 Côte de Liesse
Lachine, Québec, Canada H8T 1A1

"Toute reproduction, intégrale ou partielle sans le consentement de l'auteur, ou de ses ayants-droits, ou ayants-cause, est illicite (loi du 11 mars 1957, alinéa 1 de l'article 40).
Cette reproduction, par quelque procédé que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 et suivants du Code Pénal.
Toute reproduction au mépris de ces textes, de tout ou partie de cette documentation, entraînerait de facto, la reproduction de tout ou partie du programme informatique associé.

Introduction

LOGO est un langage.

LOGO est un langage de programmation pour communiquer avec les ordinateurs.

LOGO est un langage simple immédiatement utilisable qui se développe au fur et à mesure de votre intérêt, de votre désir, de votre compréhension.

LOGO ne vous dit jamais "NON" mais vous indique ce qu'il ne comprend pas, ce que vous ne lui avez pas encore appris.

LOGO est un langage puissant, aux caractéristiques sophistiquées. En vous familiarisant avec les notions de récursivité, de procédure, de liste, vous disposerez d'un langage de programmation moderne et complet.

Ce manuel est une initiation à la programmation Logo, il vise à introduire les idées et méthodes principales.

Par le graphique de la Tortue qui est un monde riche on peut vite comprendre, apprécier d'un seul regard le résultat de programmes, et mettre à l'épreuve les idées les plus puissantes de programmation. En outre, les résultats sont (peuvent être !) beaux.

La programmation de la Tortue est un bon support pour comprendre les méthodes du langage de programmation général qu'est LOGO. D'autre part, organiser des idées sur des mots, des listes, manipuler du texte et des nombres sont des éléments nécessaires de votre relation à l'ordinateur.

La première partie de ce manuel comporte douze chapitres ; ils constituent une initiation à la programmation LOGO. Vous pouvez les aborder même si vous n'avez jamais touché à un ordinateur auparavant.

La deuxième partie du manuel comporte deux chapitres auxquels vous pouvez vous référer à tout moment. Ils vous faciliteront l'activité de programmation et vous permettront de mieux utiliser votre ordinateur et LOGO.

Chacun des neuf premiers chapitres est divisé en trois parties de la manière suivante :

FAIRE
APPRENDRE A LA MACHINE COMMENT FAIRE
PENSER A CE QU'ON A FAIT

— vous disposez ainsi de repères qui vous permettent de lire dans n'importe quel sens ce manuel ;

— vous pouvez obtenir immédiatement des résultats sans notion préalable ;

— vous pouvez pousser votre réflexion non sur l'apprentissage d'un code, de nouveaux mots mais sur l'organisation de vos idées en fonction de vos projets, sur le sens du pourquoi et du comment de certaines conventions.

Un projet traverse ce livre : il part d'une maison et arrive à un village, en utilisant progressivement les notions introduites. De nombreux programmes ont été utilisés par ailleurs, qui vous fourniront peut-être, des idées utiles pour vos propres projets.

Il n'est pas nécessaire de tout lire pour débuter ou initier de jeunes enfants. Nous avons signalé par un signe spécial (*) les parties indispensables pour commencer ; les autres peuvent être lues dans des étapes ultérieures.

D'utilisateur "presse-bouton" LOGO vous permet d'apprendre, et de devenir créateur... Alors allumez votre ordinateur... et faites.

Table des matières

Chacun des neuf premiers chapitres est divisé de la manière suivante :

FAIRE
APPRENDRE A LA MACHINE COMMENT FAIRE
PENSER A CE QU'ON A FAIT

Les titres dans cette table des matières correspondent à la partie PENSER A CE QU'ON A FAIT de chaque chapitre.

Mise en route 10
Le clavier 12

Première partie

Chapitre 1 : Premiers contacts
1. Le symbole d'invite 21
2. La touche ENTREE 21
3. Bien écrire les instructions 22
   a) L'espace 22
   b) Succession d'instructions 22
   c) Fautes de frappe 22
4. Les "messages Logo" 23
5. L'écran 24
6. Rencontre avec la tortue 24
7. Les procédures 25
   a) Les parties d'une procédure 25
   b) Utilisation des procédures 26

Chapitre 2 : Réflexions autour d'un dessin
1. Bâtir une maison 37
2. Les procédures "briques" 37
3. L'avantage des briques 38
4. Attention à la position de la tortue ! 38
5. Les procédures récursives 38

Chapitre 3 : La tortue et son environnement
1. Son crayon 49
2. Son champ 50
   a) Comment se repérer 50
   b) La forme du champ 51
   c) La couleur du champ 52
3. L'écran graphique 52
4. La tortue dans son champ 53

Chapitre 4 : Procédures avec données
1. La donnée 61
2. Procédures avec données 62
   a) Écriture 62
   b) Exécution 63
3. Procédures imbriquées 63
4. Opérations arithmétiques sur les données 64

Chapitre 5 : De beaux dessins
1. Les tests d'arrêt 75
2. La récursivité non terminale 76

Chapitre 6 : Des mots, des noms et des instructions... définitions et écriture
1. Les mots, les noms, les listes et les ordres 80
2. Les noms et les procédures 80

Chapitre 7 : Chiffres et Lettres
1. Notion d'opération 91
2. Les expressions conditionnelles 91
3. Du rôle de PREM et SP 92

Chapitre 8 : Des procédures de dialogue
1. La primitive LL 99
2. La primitive LISCAR 100
3. La primitive MD 100

Chapitre 9 : Les procédures - opérations
1. La primitive RENDS 108
2. Les parenthèses 108
3. La récursivité dans les procédures-opérations 108
4. Les prédicats et les opérateurs logiques 109

Chapitre 10 : Le crayon optique et les manettes de jeux
1. Le crayon optique 111
2. Les manettes de jeux 114

Chapitre 11 : La musique 116

Chapitre 12 : Quatre projets
Projet 1 : Le portrait chinois 121
Projet 2 : Promenons la tortue 126
Projet 3 : ...Et la maison ! devient un paysage 129
Projet 4 : La tortue et l'oiseau 134

Deuxième partie

Chapitre 13 : Espace de travail et sauvegarde des productions
1. L'espace de travail 141
   a) Afficher 141
   b) Effacer 141
2. Sauvegarder votre travail 142
   a) Enregistrer sur cassette 142
   b) Enregistrer sur disquette 144
3. Ramener votre travail 145
   a) Cassette 145
   b) Disquette 146
4. Faire le ménage dans vos fichiers 146
5. L'imprimante 147

Chapitre 14 : Le monde de l'éditeur
1. Entrer-Sortir de l'Éditeur 148
2. Manipulations du curseur dans le monde de l'éditeur 149
3. Éditeur et procédures 150
   a) Définir une procédure 150
   b) Corriger une procédure définie en dehors de l'éditeur 151
   c) Modifier une procédure pour en faire une nouvelle 152
   d) Éditer un nom, éditer un groupe de procédures 152
4. Éditeur et écriture 153
5. Sauvegarder le contenu de l'éditeur 154
6. Imprimer le contenu de l'éditeur sur papier 155

Index 157
Liste des primitives LOGO 159

Mise en route

Pour utiliser Logo avec votre ordinateur :

1. Mettez la cartouche Logo dans la trappe qui se trouve sur la gauche du clavier.

2. Connectez et allumez tous vos périphériques (téléviseur, lecteurs, imprimante, etc.)

3. Allumez votre ordinateur.

4. Si vous avez un MO5, Logo est alors à votre disposition.
Sinon, le menu suivant apparaît :

1 Logo
2 Un programme enregistré (1)
3 Réglage du crayon optique

Appuyez sur la touche 1 pour Logo. Si vous avez un lecteur de disquettes, reporter-vous au chapitre 13.2 pour des consignes spécifiques de démarrage.

En haut de l'écran apparaît :

```
LOGO 1.Ø (c) SOLI 1984
?_
```

Ce **message** (2) indique que Logo se trouve dans la mémoire de votre ordinateur ; nous disons que Logo est chargé en mémoire.

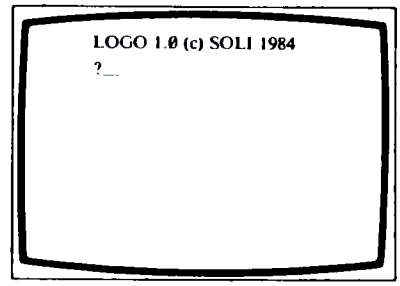
---

(1) Ce choix n'est offert que dans le cas où un lecteur de cassettes ou de disquettes est connecté.
(2) Si vous n'obtenez pas ce message :
— éteignez tout
— vérifiez que chaque appareil est bien connecté
— répétez les instructions pour charger le langage.
Si vous n'obtenez toujours rien... consultez votre distributeur.

Un point d'interrogation (?) s'affiche à gauche de la prochaine ligne. Nous l'appelons le **symbole d'invite.** Il signifie que l'ordinateur attend vos instructions.

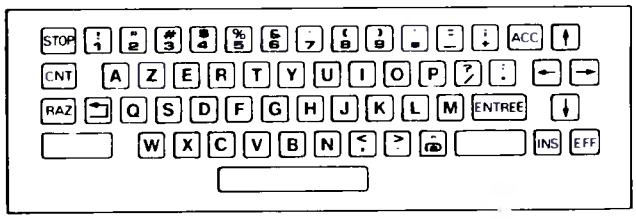
Le trait (_) qui clignote est le **curseur.** Il indique la position du prochain caractère que vous taperez.

Vous pouvez maintenant commencer votre programmation Logo...



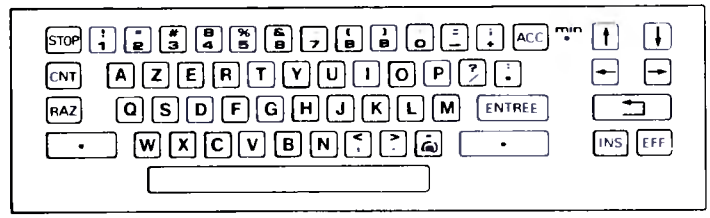
Le clavier

Le clavier comporte des touches alphanumériques (les caractères alphabétiques et numériques que vous connaissez) comme une machine à écrire avec en plus certaines touches spécifiques à l'ordinateur que nous présenterons brièvement ici.



Clavier MO5

Clavier TO7
et TO7-70



(*) 1. ENTREE Conclut une ligne et provoque sa prise en compte.

(*) 2. [ ] Barre d'espacement. Déplace le curseur en laissant un espace (appelé le blanc).

(*) 3. ← Déplace le curseur vers la gauche en effaçant le caractère.

4. CNT CONTROLE, utilisée simultanément avec une autre touche permet d'en modifier le sens.
La touche CNT permet d'obtenir les crochets
CNT A pour [
CNT Z pour]

5. STOP Interrompt momentanément une exécution. Pour la reprendre, appuyez sur n'importe quelle touche du clavier.

6. Dans l'éditeur
   - ← déplace le curseur vers la gauche
   - → déplace le curseur vers la droite
   - ↑ déplace le curseur vers le haut
   - ↓ déplace le curseur vers le bas

7. INS Dans l'éditeur, cette touche, en combinaison avec d'autres, donne accès à différentes fonctions.

8. EFF Dans l'éditeur, efface le caractère qui se trouve à la place du curseur.

9. [•] Permet d'obtenir les caractères jaunes de certaines touches du clavier. Par exemple, (!) ou (#) etc.
   Appuyez simultanément sur le point situé au milieu de cette touche et sur le caractère désiré. Attention ! Si vous possédez un MO5, cette touche est jaune, sans point dessus.

L'appui simultané sur [•] et [ ] vous permet d'écrire en lettres minuscules. Ceci est signalé par l'allumage du voyant min sur TO7 et TO7-70. Pour remettre les majuscules appuyez une deuxième fois sur ces deux mêmes touches.

10. ACC Appuyez sur [ACC] quand vous voulez mettre un accent. Ceci est possible uniquement pour les lettres minuscules. Après avoir appuyé sur cette touche vous ne voyez plus le curseur ; il restera pourtant sur la même place quand vous taperez l'accent ([•] et [accent] simultanément). Il faut taper l'accent avant la lettre à accentuer. La lettre que vous taperez ensuite se placera sous l'accent.
    Pour écrire un ç utilisez la touche [ACC] et appuyez deux fois sur la touche [C]. Si vous possédez un TO7-70 ou un MO5 les minuscules accentuées les plus courantes sont obtenues en appuyant sur :
    [ACC] puis [6] pour é
    [ACC] " [7] pour è
    [ACC] " [8] pour ù
    [ACC] " [9] pour ç
    [ACC] " [0] pour à

Les instructions Logo s'écrivent avec des lettres majuscules. Vous pouvez utiliser les minuscules et les accents si vous voulez écrire des textes qui ne sont pas des instructions du langage. Par exemple dans l'éditeur.

Première partie

Chapitre 1

Premiers contacts

Pour apprendre le Logo, il faut essayer !
Vous venez de charger le langage ; l'ordinateur attend vos instructions. Lorsque vous les aurez tapées, appuyez sur la touche, ENTREE pour qu'il les exécute. Dans le premier chapitre, cette action vous sera indiquée à chacune de ses utilisations.

Si vous ne comprenez pas ce qui se passe, n'hésitez pas à consulter la partie PENSER A CE QU'ON A FAIT de ce chapitre ; vous y trouverez une explication.

Faire

Écrivons un mot,

```
?ECRIS "BONJOUR    ENTREE
BONJOUR
```

pour le guillemet appuyez sur [•] et sur la touche [2] en même temps ; il doit figurer accollé au mot.

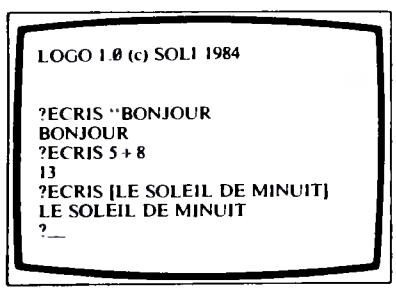
Demandons d'afficher le résultat d'un calcul,

```
?ECRIS 5 + 8    ENTREE
13
```

laissez un espace entre les nombres et le signe + et entre ECRIS et le reste

Donnons une phrase,

```
?ECRIS [LE SOLEIL DE MINUIT ]    ENTREE
LE SOLEIL DE MINUIT
```



pour [appuyez sur la touche CNT et tapez A simultanément (en maintenant un doigt sur CNT). Pour] il faut appuyer sur CNT et Z simultanément.

Appelons la tortue :

(*) ?MT ENTREE

Et l'ordinateur, Montre la Tortue.
Le triangle qui apparaît au milieu de l'écran est ce qu'on appelle la **tortue.** Votre écran se sépare en deux parties. L'écriture se fait dans la partie inférieure noire.

Voilà ! En Logo les domaines d'activité sont le graphique avec la tortue, l'arithmétique, les mots et les listes. Nous verrons par la suite les différentes possibilités dans chacun des domaines.

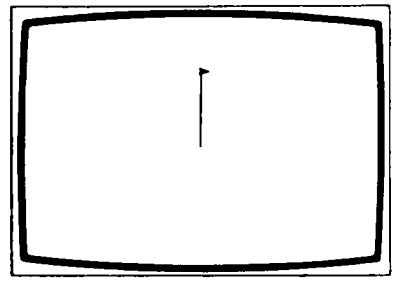
Continuons avec la tortue :

(*) ?AVANCE 80 ENTREE

En abrégé AV 80
La tortue se déplace de 80 pas en traçant une ligne. N'oubliez pas de laisser un espace entre le mot et le nombre, il est indispensable, comme si vous l'écriviez sur le papier.

Faisons la Tourner à Droite,

(*) ?TD 87 ENTREE



La tortue pivote sur sa droite de 87 degrés.

(*) ?RECULE 49 ENTREE

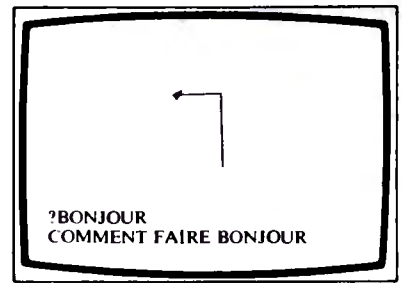
En abrégé RE 49
La tortue fait marche arrière de 49 pas.

Faisons la Tourner à Gauche,

(*) ?TG 72 ENTREE

La tortue pivote sur sa gauche de 72 degrés.

?BONJOUR [ENTREE]
COMMENT FAIRE BONJOUR



Écrit ainsi, BONJOUR est une commande comme si on disait : "FAIS BONJOUR!" Au début, la machine ne connaît pas cet ordre, mais par la suite on pourra le lui apprendre.

(*) ?VE [ENTREE]

Et l'ordinateur Vide l'Écran. Les traces disparaissent et la tortue se retrouve au milieu de l'écran.

Vous pouvez, si vous le voulez bien, recommencer. En "pilotant" la tortue vous pouvez dessiner.
Quand l'ordinateur répond immédiatement aux ordres donnés on se trouve en **mode direct.**

Apprendre à la machine comment faire

Nous avons présenté quelques termes du vocabulaire Logo comme AVANCE ou ECRIS. Ces instructions initiales s'appellent des "procédures primitives", ou par abus de langage, des **primitives.** BONJOUR n'est pas une primitive. Ce n'est pas encore une instruction Logo.

Mais elle peut le devenir !

Un des aspects intéressants de Logo est la possibilité de définir de **nouvelles instructions.** Pour ceci, vous allez définir des procédures utilisateur, appelées par abus de langage **procédures.**
Une procédure définie peut être utilisée de la même façon qu'une primitive.

1. Apprenons à la machine le mot BONJOUR

Voici ce qu'elle devrait faire :

Pour faire BONJOUR, on aimerait que la tortue salue ! Qu'elle se MONTRE, AVANCE et RECULE puis Tourne à Droite. Ceci 3 fois de suite et qu'apparaisse le message JE VOUS SALUE.

Voici comment l'écrire :

```
POUR BONJOUR
MT
AVANCE 30 RECULE 30 TD 45
AVANCE 30 RECULE 30 TD 45
AVANCE 30 RECULE 30 TD 45
ECRIS [JE VOUS SALUE]
FIN
```

CNT A pour [et CNT Z pour]

Quand vous recopiez cette procédure, remarquez qu'après avoir tapé :

?POUR BONJOUR ENTREE

Le symbole d'invite de la prochaine ligne n'est plus le ? mais devient >. Ceci indique que la machine "écoute", qu'elle ne "fait" pas.
On ne se trouve plus en "mode direct" mais en "mode de définition de procédure". La machine enregistre les instructions et les garde groupées jusqu'à ce que la procédure soit terminée.

Après avoir tapé

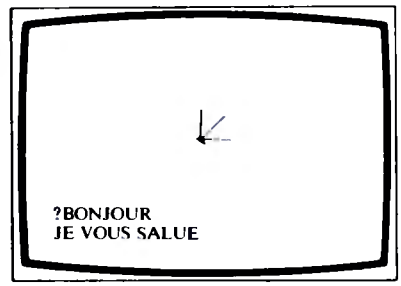
>FIN ENTREE

s'affiche sur l'écran le message

VOUS VENEZ DE DEFINIR BONJOUR

et sur la prochaine ligne, réapparaît le ?. On est à nouveau en mode direct.
Demandons :

?BONJOUR ENTREE



Note :
Pour corriger une faute de frappe :
— Si vous n'avez pas encore appuyé sur la touche ENTREE, déplacez le curseur vers l'arrière.
— Sinon, vous pouvez vous reportez au chapitre sur l'éditeur, ou terminer en tapant :
>FIN ENTREE
?EFP "BONJOUR ENTREE (attention aux guillemets)
ceci efface la procédure dont vous venez de commencer la définition.
Et ensuite vous pouvez la reécrire.

Et voilà ! la tortue va et vient, trois fois, comme prévu et la phrase JE VOUS SALUE s'inscrit sur l'écran (1).

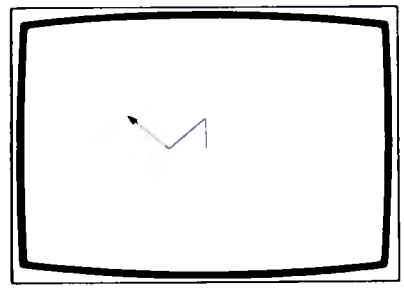
```
?VE  ENTREE
```

2. Apprenons lui autre chose

```
POUR UNTRAJET                    ENTREE
AVANCE 35                        ENTREE
TD 45                            ENTREE
RECULE 50                        ENTREE
TG 98                            ENTREE
AVANCE 47                        ENTREE
FIN                              ENTREE
```

Essayons UNTRAJET pour voir ce qui apparaîtra.

```
?UNTRAJET  ENTREE (2)
?VE  ENTREE
```



3. Essayons de faire un escalier

Au départ, la tortue pointe vers le haut de l'écran.
Pour faire un escalier, elle devra AVANCEr de 20 pas, Tourner à Droite de 90 degrés, et AVANCEr encore de 20 pas pour faire la marche.

---

(1) Si vous n'obtenez pas ceci, vous avez dû faire erreur dans l'écriture de la procédure. Deux solutions :
a) Effacer la procédure :

```
?EFP "BONJOUR
```

et recopier à nouveau la procédure BONJOUR correctement...
b) Éditer la procédure, comme vous l'avez écrite, et la corriger.
Reportez-vous au chapitre sur l'éditeur. Cette méthode est la plus économique.

(2) Le nom d'une procédure est toujours écrit en un seul mot. Si en français UNTRAJET s'écrit en plusieurs mots, en Logo il faut écrire les lettres les unes à la suite des autres pour en faire un seul.

Elle doit ensuite Tourner à Gauche de 90 degrés pour commencer la marche suivante.

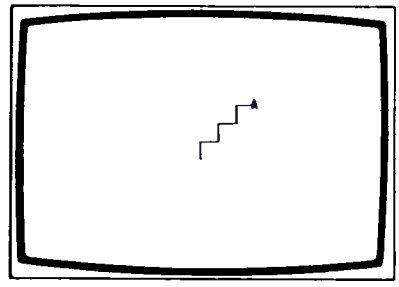
Faisons un escalier de trois marches :

```
POUR ESCALIER                    ENTREE
AVANCE 20 TD 90                  ENTREE
AVANCE 20 TG 90                  ENTREE
AVANCE 20 TD 90                  ENTREE
AVANCE 20 TG 90                  ENTREE
AVANCE 20 TD 90                  ENTREE
AVANCE 20 TG 90                  ENTREE
FIN                              ENTREE
```

Essayons-le :

?ESCALIER ENTREE

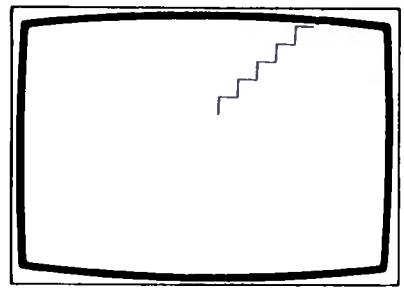
Est-ce bien ?



Faisons un deuxième ESCALIER qui monte, qui monte ... Pour chaque marche, la tortue devra refaire les mêmes actions : avancer, tourner, dans un sens, avancer, tourner dans l'autre sens.

```
POUR ESCALIERM                   ENTREE
AVANCE 20 TD 90                  ENTREE
AVANCE 20 TG 90                  ENTREE
ESCALIERM                        ENTREE
FIN                              ENTREE
```

?ESCALIERM ENTREE



Après la cinquième marche la tortue s'arrête et le message suivant apparaît : DANS ESCALIERM, LA TORTUE VA SORTIR

Pour le moment, la tortue ne peut pas "sortir" de l'écran pour continuer son dessin !

Penser à ce qu'on a fait

Nous avons fait tellement de choses ! Arrêtons-nous un moment pour y réfléchir.

1. Le symbole d'invite

Le point d'interrogation en début de ligne (?) signifie que vous vous trouvez en **mode direct**. La machine attend vos instructions pour les exécuter immédiatement.

Le signe > en début de ligne signifie que vous êtes en **mode de définition de procédure.** La machine "écoute" vos instructions ; vous définissez une procédure utilisateur.

2. La touche ENTREE

Cette touche est importante pour la communication avec la machine. **En mode direct,** elle conclut la ligne Logo (1) en cours et la fait "entrer" dans l'ordinateur. Celui-ci "reçoit" vos instructions et les exécute ou vous envoie un message Logo. En appuyant sur ENTREE c'est comme si vous disiez à la machine : "A TOI ! FAIS CE QUE JE DEMANDE". Le curseur se retrouve à la ligne suivante, précédé du symbole d'invite ?. De nouvelles instructions sont attendues.

(1) Notez qu'une ligne Logo peut s'étendre sur plusieurs lignes d'écran. Elle se termine par l'appuie de ENTREE et peut avoir en mode direct jusqu'à 248 caractères (espaces inclus).

En mode de définition de procédure, la touche ENTREE conclut la ligne Logo et met le curseur au début de la ligne suivante. Les instructions ne seront pas exécutées mais stockées en mémoire. L'exécution d'instructions n'est pas possible tant que la définition de la procédure est en cours. Elle le devient lorsque le point d'interrogation réapparaît en début de ligne.

3. Bien écrire les instructions

Une instruction est "un travail demandé" à la machine.

a) L'espace

Un mot du vocabulaire Logo est, comme dans toutes les langues, délimité par deux espaces. La machine "parcourt" la ligne écrite de gauche à droite, s'arrête quand elle rencontre un espace et essaye d'exécuter. Si pour l'exécution elle a besoin d'une information, elle la recherche dans le mot suivant.

Faites attention de toujours laisser un espace entre une commande et un nombre, comme par exemple dans AVANCE 80, ainsi qu'entre les différentes instructions.

b) Succession d'instructions

Sur une ligne Logo, vous pouvez écrire plusieurs instructions les unes à la suite des autres. Par exemple :
AVANCE 56 TD 79 AVANCE 138 TD 40 RECULE 59 ECRIS "OUF
Cependant, il faut que chaque instruction soit complète et un espace doit la séparer de la suivante. En appuyant sur la touche ENTREE la machine les exécutera les unes après les autres dans le sens de l'écriture.

c) Faute de frappe

En mode direct, si vous avez fait une faute de frappe corrigez-la avec la touche ←. *Si vous avez déjà appuyé sur* ENTREE, alors, la machine affichera un message au lieu de faire ce que vous auriez voulu.
Pour rectifier, il faut réécrire sur la ligne suivante l'instruction entière correctement. On ne peut pas faire de corrections sur une ligne qui a déjà été prise en compte.

En mode de définition de procédure, pour corriger une ligne déjà prise en compte (par l'appui de la touche ENTREE) reportez-vous au chapitre sur l'éditeur.

4. Les "messages Logo"

Parfois la machine affiche un "message Logo"
Par exemple, si vous avez tapé :

```
?AVANCE  [ENTREE]
PAS ASSEZ DE DONNEES POUR AVANCE
```

La machine vous rappelle que vous ne lui avez pas indiqué le nombre de pas dont la tortue doit avancer.

Si vous tapez :

```
?80  [ENTREE]
QUE FAIRE DE 80
```

Un nombre est normalement un quelque chose qui accompagne une action. On doit indiquer cette action à la machine : avancer, tourner, écrire, reculer etc. Il ne peut être employé seul. Il faut donc réécrire l'instruction complète.

Si vous tapez :

```
?AVANCE80  [ENTREE]
COMMENT FAIRE AVANCE80
```

La machine lit la suite de caractères comme un mot nouveau ; mot dont elle ne connaît pas la signification. Il faut réécrire l'instruction en séparant le nombre de l'action par un espace.

Si vous avez demandé un grand nombre de pas :

```
?AVANCE 1000  [ENTREE]
LA TORTUE VA SORTIR
```

Pour l'instant la tortue ne peut pas dépasser les bornes de l'écran.

Après avoir conclu la procédure BONJOUR par FIN, la machine vous annonce :

```
VOUS VENEZ DE DEFINIR BONJOUR
```

BONJOUR fait alors partie du vocabulaire Logo.

Il est important de bien lire ces messages car ils vous donnent de précieuses indications. Il s'agit d'un dialogue entre la machine et vous.

5. L'écran

Au démarrage, vous pouvez écrire sur toute la surface de votre écran. Pour vous permettre d'avoir simultanément le "dessin" et le texte, l'écran devient mixte lorsque vous utilisez la tortue. On écrit sur les quatre lignes du bas. Vous pouvez ainsi passer d'un écran texte, à un écran graphique ou mixte, sur le même téléviseur.

La primitive ME (Mixe Ecran) vous permet de délimiter le nombre de lignes que vous voulez utiliser pour le texte. ME 1 vous permet d'avoir une seule ligne pour l'écriture (au bas de l'écran) ME 25 vous donne 25 lignes d'écriture (la totalité de l'écran).
Mais **attention !** Quand vous passez d'un écran texte à un écran graphique, (ou l'inverse), **ce qui apparaissait précédement à l'écran n'est pas conservé.**

6. Rencontre avec la tortue

Vous avez déjà fait connaissance avec la tortue ! C'est un "robot" symbolisé par un triangle. Elle se déplace dans les directions et sens de sa pointe ou pivote autour de son axe. Elle possède un crayon avec lequel elle peut laisser une trace de son passage. En la guidant, on peut obtenir différents dessins.

Vous avez pu remarquer que la tortue ne fait qu'une chose à la fois. Elle se déplace ou elle pivote.
Elle exécute les instructions que vous lui donnez les unes après les autres, et chaque instruction doit être complète.
Si on écrit :

```
AVANCE TD 30
```

La machine répondra :

```
PAS ASSEZ DE DONNEES POUR AVANCE
```

La tortue tourne de 30 degrés à droite mais n'avance pas de 30 pas comme on pourrait imaginer. Chaque instruction doit ici avoir sa propre information (son propre nombre).

Il faut toujours préciser le nombre de pas dont la tortue avance ou recule et le nombre de degrés dont elle pivote (un tour complet comporte 360 degrés).

7. Les procédures

Les procédures vous permettent de faire réaliser de nouvelles tâches à la machine.

a) Les parties d'une procédure

Un exemple :

```
POUR BONJOUR
MT
AVANCE 30 RECULE 30 TD 45
AVANCE 30 RECULE 30 TD 45
AVANCE 30 RECULE 30 TD 45
ECRIS [JE VOUS SALUE]
FIN
```

le titre

le corps de la procédure comportant des instructions

fin de la définition

BONJOUR est maintenant un nouveau mot du vocabulaire Logo.

— Première ligne : POUR BONJOUR
POUR est toujours le premier mot du titre. Il indique qu'une définition suit ; il est suivi du nom de la procédure (écrit en un seul mot). C'est comme si vous disiez à la machine : "POUR faire BONJOUR voici la recette". En appuyant sur ENTREE le symbole d'invite de ? devient >. La machine écoute et enregistre ce que vous avez à lui dire.

— Le corps de la procédure.
Ce sont les instructions qui constituent la definition. Cette partie peut contenir plusieurs lignes avec des mots connus du langage : des primitives ou des procédures déjà définies.

— Dernière ligne : FIN
La dernière ligne contient uniquement le mot FIN ; il indique que cette définition est terminée.

Le message Logo : VOUS VENEZ DE DEFINIR BONJOUR est affiché pour annoncer que ce mot fait désormais partie du vocabulaire Logo.

On est à nouveau en mode direct : ? apparaît en début de ligne.

Pour que la procédure qui vient d'être définie soit exécutée, il faut le demander. Chaque fois qu'on demandera BONJOUR, l'ordinateur fera ce que nous lui avons "enseigné".

b) Utilisation des procédures

Dès qu'une procédure utilisateur est définie elle est équivalente à une procédure primitive. On peut l'utiliser aussi bien en mode direct que dans la définition d'autres procédures.

Les procédures restent en mémoire tant que l'ordinateur est allumé. Attention ! si vous l'éteignez il aura oublié tout ce que vous lui avez appris.

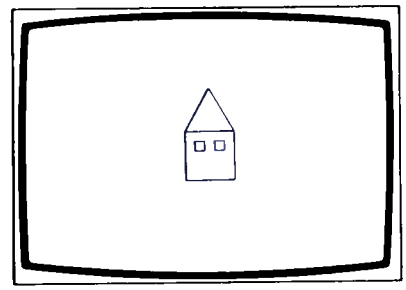
Pour sauvegarder vos procédures et ensuite les retrouver, reportez-vous à la deuxième partie de ce manuel, Chapitre 13. Ainsi, en retrouvant vos productions vous disposerez d'un langage de plus en plus puissant, enrichi par chacun de vos nouveaux apports.

Chapitre 2

Réflexions autour d'un dessin

Dans ce chapitre, nous aborderons l'utilisation des procédures pour mener à terme un projet.

Le projet est le suivant : tracer une maison



Par la suite, vous retrouverez dans tous les chapitres, sous la rubrique ...Et la maison !, ce dessin enrichi par les nouvelles notions introduites. Il deviendra ainsi progressivement un village et un paysage.

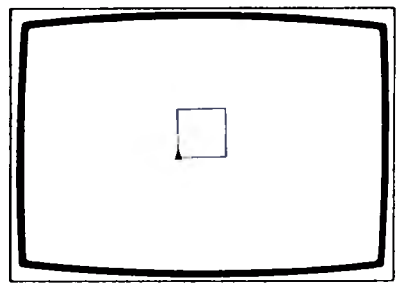
A partir de maintenant, nous n'indiquerons plus l'action sur la touche ENTREE , à la fin de chaque ligne Logo

Faire

Nous allons guider la tortue pour faire la maison.

?AVANCE 50
?TD 90
?AVANCE 50
?TD 90
?AVANCE 50
?TD 90
?AVANCE 50
?TD 90 pour la remettre dans la position de départ.

Voici la façade !



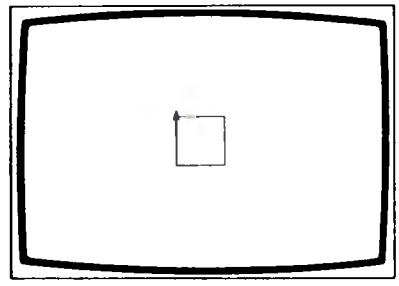
Tout ceci aurait pu s'écrire :

```
?REPETE 4 [AVANCE 50 TD 90]
```

REPETE est une primitive utile. Au lieu de réécrire les mêmes instructions, il suffit d'**indiquer** à la machine le **nombre** de fois qu'il faut répéter **la liste d'instructions** écrite entre **crochets.**

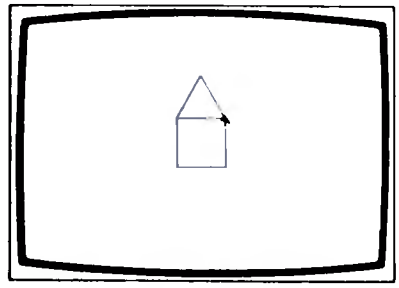
Mettons le toit :

```
?AVANCE 50
```



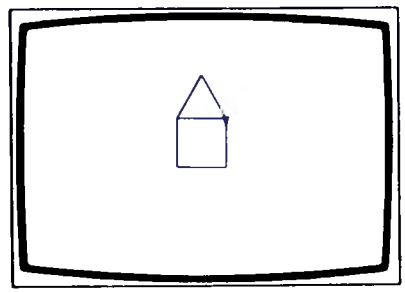
Elle se trouve en haut de la façade. Faisons-la pivoter un peu sur sa droite pour tracer le toit.

```
?TD 30
?AVANCE 50
?TD 120
?AVANCE 50
```



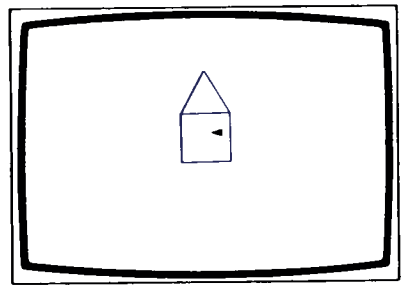
Dessinons les fenêtres :
Plaçons d'abord la tortue sur le mur de façade.

```
?TD 30
?LC
```



La tortue Lève son Crayon ; désormais elle ne laissera plus de trace dans ses déplacements.

```
?AVANCE 20 TD 90
?AVANCE 10
```

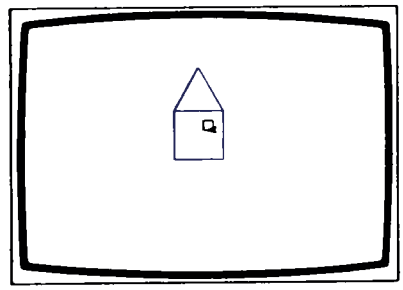


Elle se trouve à l'intérieur de la maison et on peut maintenant dessiner une fenêtre :

```
?BC
```

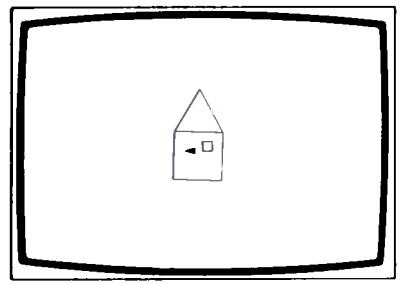
La tortue Baisse son Crayon. Au prochain déplacement elle tracera.

```
?AVANCE 10 TD 90
?AVANCE 10 TD 90
?AVANCE 10 TD 90
?AVANCE 10 TD 90
```

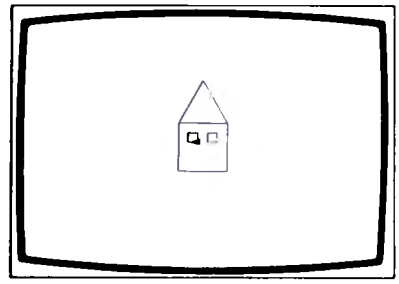


Déplaçons-la pour faire la deuxième fenêtre :

```
?LC AVANCE 10
?AVANCE 10 BC
```

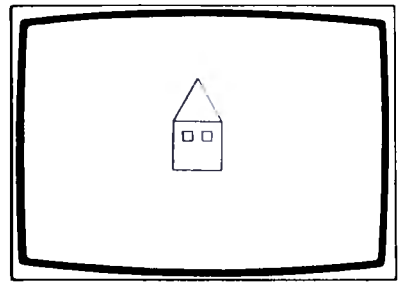


```
?REPETE 4 [AVANCE 10 TD 90]
```



```
?CT
```

Ainsi on Cache la Tortue. Et la maison tracée reste sur l'écran.



Apprendre à la machine comment faire

Pour apprendre à la machine comment faire cette maison nous avons deux possibilités :
— soit définir une procédure en reprenant les différents ordres donnés en mode direct
— soit essayer de décomposer le dessin de la maison en différentes parties.
Ainsi, on distingue :

la façade — un grand carré
le toit — un triangle
les fenêtres — deux petits carrés

Nous pouvons définir des procédures pour chacune de ces parties. Ensuite nous les assemblerons pour obtenir la maison.

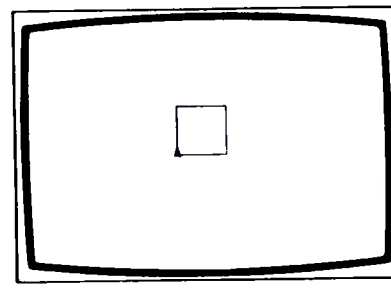
1. Bâtir la façade

Faisons le grand carré. La tortue AVANCE de 50 pas et Tourne à Droite d'un quart de tour (90 degrés) 4 fois.

```
POUR GCARRE
REPETE 4 [AVANCE 50 TD 90]
FIN
```

Demandons :

```
?VE
?GCARRE
```



Voici la façade !

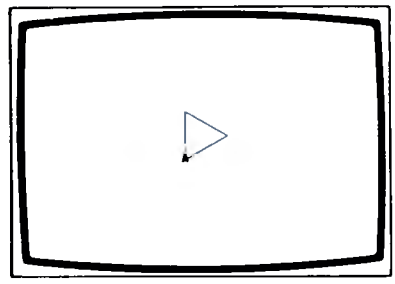
2. ... et le toit

Faisons un triangle. La tortue AVANCE de 50 pas (même longueur que le côté du carré) et pivote de 120 degrés pour se mettre dans la bonne direction afin de tracer le prochain côté. Ceci est fait 3 fois de suite.

```
POUR TRIANGLE
REPETE 3 [AVANCE 50 TD 120]
FIN
```

Essayons :

```
?VE
?TRIANGLE
```

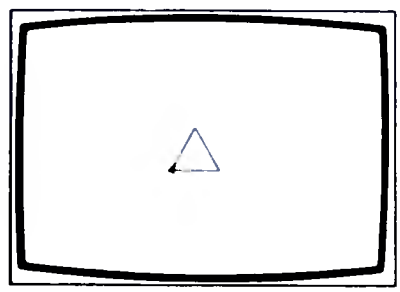


Il faudrait "pencher" un peu la tortue au départ pour que ce triangle ressemble à un toit.

```
POUR TOIT
TD 30 TRIANGLE
FIN
```

Voyons

```
?VE
?TOIT
```



Voici le toit !

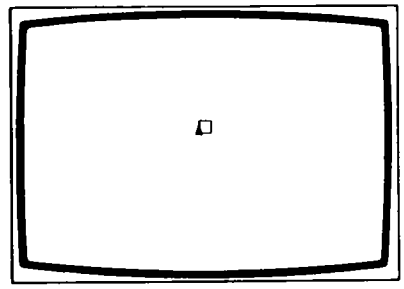
3. ... et les fenêtres

Les deux fenêtres sont identiques. Nous allons donc définir une seule procédure PCARRE que nous utiliserons deux fois pour la maison.

```
POUR PCARRE
REPETE 4 [AVANCE 10 TD 90]
FIN
```

Voyons

```
?VE
?PCARRE
```



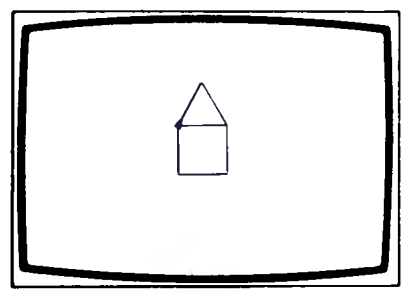
4. ...Et la maison !

Nous avons défini la façade, le toit et les fenêtres ; tout ce qui constitue cette maison. Il s'agit maintenant de les placer les uns par rapport aux autres de façon à reconstituer la maison. Il faut, par exemple, que le toit soit tracé au-dessus des murs et non à côté et que les fenêtres soient sur la façade et pas en dessous etc.
Allons-y :

```
?GCARRE
```

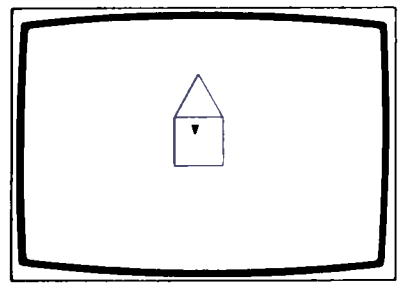
Maintenant plaçons la tortue en haut pour tracer le toit :

```
?AVANCE 50
?TOIT
```



Pour faire les fenêtres elle doit se trouver à l'intérieur du GCARRE :

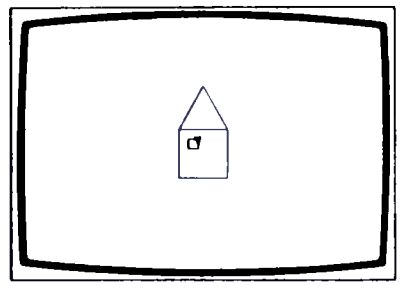
```
?TD 60
?LC AVANCE 20
?TD 90 AVANCE 10 BC
```



C'est la bonne place !

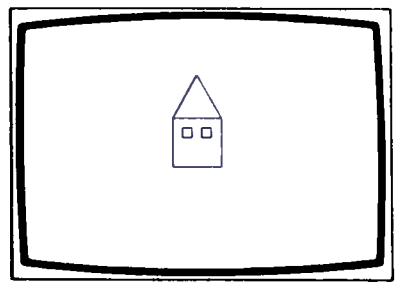
Traçons une fenêtre :

```
?PCARRE
```



Mettons la tortue à la bonne position pour tracer la deuxième fenêtre :

```
?LC TG 90 AVANCE 10 BC
?PCARRE
?CT
```



Voilà !

Écrivons maintenant la procédure :

```
POUR MAISON
GCARRE
AVANCE 50 TOIT
TD 60
LC AVANCE 20 TD 90 AVANCE 10
BC PCARRE
LC TG 90 AVANCE 10
BC PCARRE
CT
FIN
```

```
?MAISON
```

Qu'en pensez-vous ?

```
?VE
```

A suivre...

5. Et des sous-titres

Nous pouvons sous-titrer ce dessin en français !

```
POUR MAISON2
ME 6
ECRIS [POUR DESSINER UNE MAISON]
GCARRE
ECRIS [IL FAUT BATIR LES MURS]
AVANCE 50 TOIT
ECRIS [POSER LE TOIT]
TD 60
LC AVANCE 20 TD 90 AVANCE 10
BC PCARRE
ECRIS [METTRE LES FENETRES]
LC TG 90 AVANCE 10
BC PCARRE
CT
ECRIS [MAIS QUI Y HABITERA ?]
FIN
```

Essayons :

?MAISON2

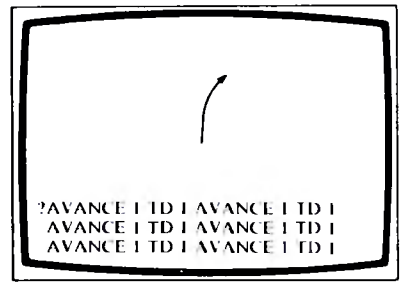
6. Et le cercle ?

Comment apprendre à la tortue à tracer un cercle ? Essayons, comme pour la maison, de définir le cercle comme une succession d'actions faciles à décrire.

Pour marcher le long d'un cercle, nous avançons en tournant.

Ces deux actions sont séparées et effectuées l'une après l'autre par l'ordinateur. Alors on peut : avancer un peu, puis tourner un peu, plusieurs fois. En fait, il suffit de savoir faire un petit bout de chemin sur un cercle : après c'est toujours pareil !

Essayons en mode direct plusieurs fois AVANCE 1 TD 1. Nous voyons effectivement un arc de cercle se dessiner sur l'écran.



Essayons de définir la procédure CERCLE comme AVANCE 1 TD 1 et recommencer c'est-à-dire reprendre les mêmes actions indéfiniment.

```
POUR CERCLE
AVANCE 1 TD 1
CERCLE
FIN
```

Demandons

?CERCLE

Pas mal !... Mais la tortue tourne en rond !!! Appuyez sur les touches CNT et C simultanément pour l'arrêter. (Gardez un doigt sur CNT pendant que vous appuyez sur C).

Penser à ce qu'on a fait

1. Bâtir une maison

Dans la première partie de ce chapitre nous avons tracé une maison en "pilotant" la tortue en **mode direct**. Mais ce que nous avons construit est définitivement perdu si nous "vidons" l'écran.

Pour pouvoir en faire un acquis et reproduire ce dessin sans tout réécrire à chaque fois, il faut définir une procédure.

Si nous voulons recopier sous forme de procédure toutes les instructions données en mode direct, nous aurons une procédure contenant un très grand nombre d'instructions ! Il sera difficile, par la suite, de retrouver à l'intérieur de celle-ci ce qui a besoin d'être corrigé ou modifié.

2. Les procédures "briques"

Dans la deuxième partie du chapitre, pour "apprendre" à la machine comment faire une maison nous avons suivi une stratégie appelée la "programmation structurée".

Nous avons décomposé le problème :
"Dessiner une maison"

en une série de problèmes :
"Dessiner la façade"
"Dessiner le toit"
"Dessiner les fenêtres"

Chacun de ces petits problèmes devient alors plus facile à comprendre et à réaliser. Nous avons pu ainsi définir des petites procédures, associées à chacune des parties de la maison. Chacune de ces procédures constitue une brique.

Pour obtenir le dessin final, nous avons simplement regroupé ces parties en une nouvelle procédure, MAISON.

3. L'avantage des briques

Quand on définit plusieurs procédures brèves, il est plus facile pour l'utilisateur de comprendre et de contrôler chacune d'elles.

Dès qu'il s'agit de faire des corrections ou des modifications, on peut directement savoir à quel endroit intervenir.
Ainsi, pour modifier une instruction, par exemple la taille de la fenêtre, il suffit de changer la valeur correspondante à l'intérieur de la procédure qui la comporte, PCARRE.

Une procédure a le même usage qu'une primitive. Elle peut être utilisée dans différents contextes. Ainsi TRIANGLE devient ici le toit pour la maison, mais la même procédure peut être utilisée pour faire, par exemple, le chapeau d'un personnage en combinaison avec CERCLE, ou le corps d'un arbre etc.

4. Attention à la position de la tortue !

Vous avez sûrement remarqué que la tortue trace une forme à partir de la position et de la direction qu'elle occupe au moment de la demande. Faites très attention, lors de l'enchaînement de formes, de l'amener au bon endroit avec le bon angle de départ avant de lui demander de tracer ce que vous voulez.

5. Les procédures récursives

Une stratégie Logo consiste à décrire la suite d'actions à effectuer de façon à :
a) Mettre en évidence des actions semblables : c'est ce qui a été ébauché dans la MAISON pour GCARRE et PCARRE et qui sera repris au Chapitre 4.

b) Mettre en évidence une suite d'actions à répéter. Le nombre de répétitions peut être fixe comme dans GCARRE ou infini comme dans le cas de l'ESCALIERM du Chapitre 1 ou de CERCLE.

Voyons ce qui se passe dans les procédures ESCALIERM ou CERCLE. Dans le cas de l'ESCALIERM la suite d'actions retenue est celle qui construit une marche. Pour le cercle la suite d'actions est celle qui construit un arc (ou du moins une approximation...)

Comment construire un escalier qui monte qui monte...

.....

une marche

une marche

une marche

une marche

Ce qui s'écrit en Logo :

.....

AVANCE 20 TD 90 AVANCE 20 TG 90

AVANCE 20 TD 90 AVANCE 20 TG 90

AVANCE 20 TD 90 AVANCE 20 TG 90

Mais on ne s'arrêtera jamais d'écrire !

La syntaxe Logo permet d'appeler ESCALIERM pendant sa définition : c'est la réalisation informatique de cette écriture impossible.

.....

.....

.....

ESCALIERM

une marche

ESCALIERM

une marche

ESCALIERM

Les procédures qui "s'appellent" elles-mêmes pendant leur exécution sont les **procédures récursives** (1).

L'appel du nom de la procédure peut se faire :

— soit directement dans le corps de la procédure comme c'est le cas pour ESCALIERM et CERCLE dont le titre réapparaît comme une instruction du corps de la procédure.

— soit à travers d'autres procédures, par exemple :

```
POUR ESSAI
AVANCE 10 TD 45
SALUT
FIN
```

---

(1) Pour parler de la récursivité, on utilise souvent l'histoire suivante : une fée promet à quelqu'un de réaliser deux de ses souhaits. Le deuxième désir de cette personne est toujours de pouvoir encore formuler deux souhaits !

```
POUR SALUT
ECRIS "BONJOUR
RECULE 10 TG 45
ESSAI
FIN
```

ESSAI appelle SALUT qui appelle ESSAI qui appelle SALUT... Pour arrêter l'exécution d'une telle procédure récursive tapez sur les touches CNT et C simultanément, mais comme nous le verrons au Chapitre 5, ce n'est pas la seule possibilité !

Chapitre 3

La tortue et son environnement...

Ce chapitre vous aide à explorer le monde de la tortue.
Nous aborderons ici le champ de ses déplacements, les repères géographiques, et l'utilisation de la couleur.

Faire

```
?AVANCE 50
```

Ce dernier trait ne me plaît pas. Vite effaçons-le !

```
?FCC 4
```

Ce qui veut dire : Fixe la Couleur du Crayon à la couleur 4. La tortue prend la couleur du fond et n'est plus visible.

```
?RECULE 50
```

Le trait est effacé.
Et pour tracer à nouveau :

```
?FCC 1
```

Fixons la Couleur du Crayon à 1

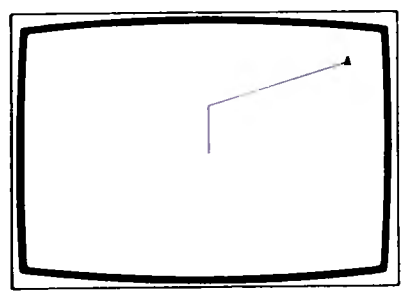
```
?AVANCE 50
```

Le tracé est rouge.

```
?ECRIS CC
1
```

CC nous donne le code de la Couleur du Crayon de la tortue.
Déplaçons la tortue jusqu'au coin supérieur droit :

```
?FPOS [150 90]
```



Elle se pose près du coin droit de l'écran en laissant une trace et sans changer son orientation.

```
?FPOS [-150 90]
```

Déplace la tortue horizontalement vers le bord gauche. Pour indiquer un nombre négatif, il faut accoler le signe – au chiffre.

```
?FPOS [0 0]
```

Ramène la tortue au centre.
Tournons-lui la tête :

```
?FCAP 90
```

La tortue regarde vers la droite.

```
?FCAP 180
```

Maintenant elle regarde vers le bas.

Un peu d'exercice :

```
?VE AVANCE 50 TD 45 RECULE 30 TG 27 AVANCE 20 TG 170
```

Quelle est la position de la tortue sur l'écran ? Vers où regarde-t-elle ?

```
?ECRIS POS
-15      47.7
```

POS nous donne la position de la tortue sur l'écran.

```
?ECRIS CAP
208
```

CAP nous donne la direction vers laquelle la tortue "regarde" exprimée par l'angle qu'elle fait par rapport au nord (haut de l'écran).

Jusqu'où peut-elle se déplacer ?

```
?VE AVANCE 1000
LA TORTUE VA SORTIR
```

Quelle que soit son orientation, le résultat est le même, la tortue est prisonnière dans les limites de l'écran, son champ est un CLOS.

```
?FEN
```

Equivaut à FENêtre

```
?AVANCE 500
```

La tortue a disparu !

Même résultat quelle que soit son orientation.

```
?VE
?ENR
```

Equivaut à ENRoule

```
?AVANCE 500
```

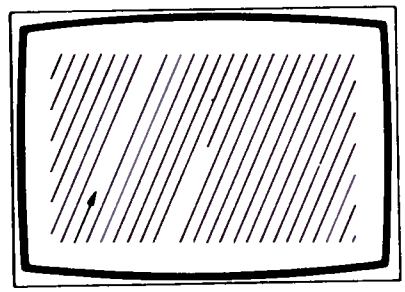
La tortue est sortie en haut de l'écran pour réapparaître sur la même verticale en bas...

```
?VE TD 90 AVANCE 500
```

La tortue est sortie à droite et est revenue à gauche sur la même horizontale...
ou encore :

```
?VE TD 25 AVANCE 5000
```

L'écran se remplit de zébrures !



```
?FCFG 5
```

Fixe la Couleur du Fond Graphique à la couleur 5. Tout l'écran devient violet (magenta).

```
?ECRIS CF
5
```

CF nous donne le code de la couleur du fond.

Voici donc des primitives qui vont nous permettre

— de faire de beaux effets de couleur
— de positionner la tortue sur l'écran
— de jouer avec la forme du champ pour autoriser la tortue à partir de l'écran (FEN), revenir (ENR) ou rester dans les limites de l'écran (CLOS).

Apprendre à la machine comment faire

1. Effets colorés

Reprenons l'effet de zébrures que donnent dans l'état ENR les instructions TD 25 AVANCE 5000. Nous allons faire changer la couleur du crayon chaque fois que la tortue aura parcouru 100 pas.
Par l'appel récursif ceci se répétera indéfiniment.

```
POUR EFFET
AVANCE 100 FCC CC + 1
EFFET
FIN
```

Essayons :

```
?ENR BC TD 30 EFFET
```
CNT C pour arrêter

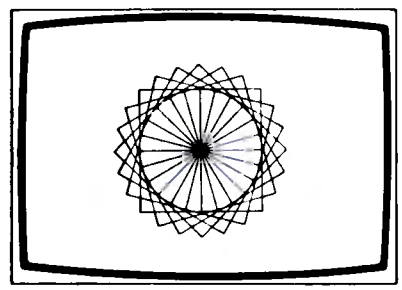
Quel bel effet de couleurs !!!

2. Carré qui tourne en changeant de couleur

```
POUR TOURNEC
FCC CC + 1                          choisir la prochaine couleur
REPETE 4 [AVANCE 50 TD 90]          tracer le carré
TD 20                               tourner à droite
TOURNEC                             recommencer
FIN
```

Regardons :

```
?VE
?TOURNEC
```



CNT C pour arrêter

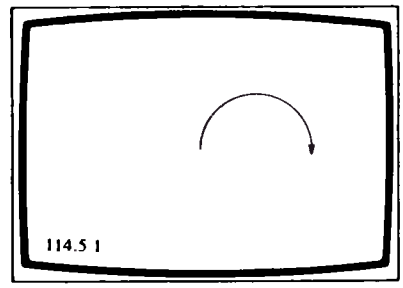
3. Le chronomètre

Ce sera tout d'abord un cercle ! Mais il faut repérer le centre pour pouvoir mettre les aiguilles.
Nous allons alors, faire :

```
POUR DIAMETRE
REPETE 180 [AVANCE 1 TD 1]
ECRIS POS
FIN
```

Réalise un demi-cercle
écrit les coordonnées du point diamétralement opposé au point de départ

```
?VE DIAMETRE
114.5  1
```



Maintenant nous avons le diamètre du cercle. Le centre se situe au milieu : 114.5/2 = 57.25
Plaçons l'aiguille au centre

```
POUR AIGUILLE
LC FPOS [57.25 0]
FCAP 0
FIN
```

dirigée vers le haut de l'écran

Et pour faire tourner l'aiguille avançons de 50 pas puis reculons d'autant en gommant ; on tourne ensuite de 6 degrés et on recommence.
Remarquons qu'une couleur de crayon donnée par un nombre négatif est une couleur pour le fond du crayon c'est-à-dire pour tout un rectangle encadrant la tortue (voir manuel de référence). C'est ce que nous allons utiliser ici pour définir une "bonne" gomme.

```
POUR TAIGUILLE
BC AVANCE 50 RECULE 50
FCC - (CF + 1)
AVANCE 50 RECULE 50
FCC 1 TD 6
TAIGUILLE
FIN
```

met le fond du crayon à la couleur du champ.

Le cadre :

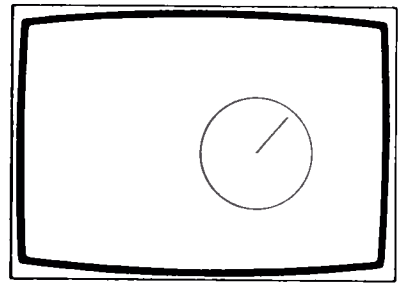
```
POUR CADRE
REPETE 360 [AVANCE 1 TD 1]
FIN
```

Et enfin le tout ensemble :

```
POUR CHRONO
CT
CADRE
AIGUILLE
FCC 1
TAIGUILLE
FIN

?CHRONO
```

CNT C pour l'arrêter



4. L'oiseau

Faisons un oiseau qui vole. Pour donner l'illusion du mouvement on peut : dessiner, effacer le dessin et refaire la même chose un peu plus loin.

a) Essayons d'abord de dessiner l'oiseau.

```
POUR OISEAU
TG 45 AVANCE 8 RECULE 8
TD 90 AVANCE 8 RECULE 8
TG 45
FIN
```

Demandons :

```
?OISEAU
```

il est prêt à s'envoler !

b) Faisons une deuxième procédure pour effacer cet oiseau ; il suffit de faire le même chemin d'OISEAU, en couleur du fond.

```
POUR EFOISEAU
FCC - ( CF + 1 ) OISEAU
FIN
```

```
?EFOISEAU
```

Et voilà !

c) Faisons le voler à présent ! Nous le ferons apparaître, disparaître et recommencer plus loin. C'est le principe du dessin animé.

```
POUR VOLE
FCC 5 OISEAU
EFOISEAU
LC AVANCE 5
BC VOLE
FIN
```

Essayons :

```
?CT VOLE
```

L'oiseau s'élève dans le ciel et s'arrête en haut de l'écran.

Faisons le voler horizontalement.

```
POUR VOLER
CT LC FPOS [-100 90] FCAP 270
VOLE
FIN

?VOLER
```

L'oiseau vole dans le ciel mais s'arrête aux limites de l'écran.

```
?ENR CT VOLER
```

Le voilà qui réapparaît. Appuyez sur CNT et C simultanément pour l'arrêter.

5. ... Et la maison !

Reprenons la procédure MAISON du Chapitre 2 et donnons-lui de la couleur : (1) faisons : EFP "MAISON puis:

```
POUR MAISON
FCFG 0
FCC 3 GCARRE
AVANCE 50 FCC 1 TOIT
TD 60
LC AVANCE 20 TD 90 AVANCE 10
BC FCC 5 PCARRE
LC TG 90 AVANCE 10
BC PCARRE
CT
FIN
```

Nous pouvons ajouter notre oiseau à la maison, et ainsi constituer une scène.

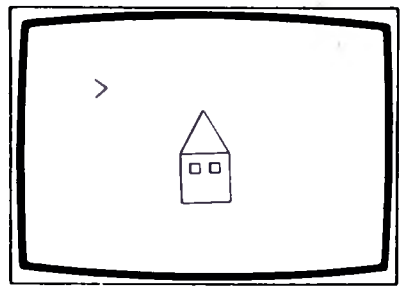
```
?VE
?ENR

POUR SCENE
MAISON
VOLER
FIN
```

---

(1) Les procédures GCARRE TOIT et PCARRE devront être recopiées ou ramenées dans la mémoire centrale si vous avez entre temps éteint votre appareil. (voir dans Chapitre 13 Sauvegarder votre travail).

```
?SCENE
```



C'est bon ! Pour arrêter, appuyez sur les touches CNT et C

A suivre...

Penser à ce qu'on a fait

Au démarrage la tortue se trouve au centre d'un champ de couleur bleu, regard tourné vers le Nord (haut de l'écran) crayon bleu ciel prêt à tracer. Les possibilités de déplacements sont délimitées par les bords de l'écran.
Les primitives que nous venons d'utiliser permettent de modifier cet état initial.

1. Son crayon

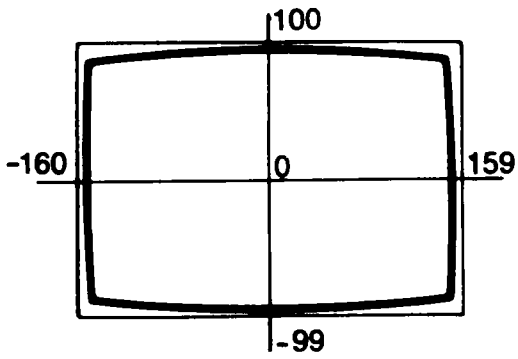
Il est caractérisé par :
— sa "situation" levé : il ne trace pas, baissé il trace.
— sa couleur suivant les codes ci-dessous. Dans le cas d'un code positif, le dessin se fera normalement. Dans le cas d'un code négatif, le crayon peut colorer tout un rectangle.

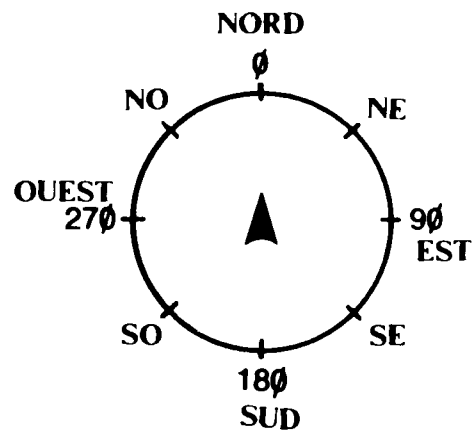
Code positif	Code négatif	Couleur
Ø	−1	NOIR
1	−2	ROUGE
2	−3	VERT
3	−4	JAUNE
4	−5	BLEU
5	−6	MAGENTA (VIOLET)
6	−7	CYAN (BLEU CLAIR)
7	−8	BLANC
8	−9	GRIS
9	−1Ø	ROUGE PÂLE
1Ø	−11	VERT PÂLE
11	−12	JAUNE PÂLE
12	−13	BLEU PÂLE
13	−14	MAGENTA PÂLE
14	−15	CYAN PÂLE
15	−16	ORANGE

pour MO5 et TO7-7Ø seulement

2. Son champ

a) Comment se repérer





On peut imaginer le champ de la tortue comme ayant un quadrillage sur lequel elle peut se déplacer, et une boussole.

L'instruction FCAP suivie d'un nombre indique l'orientation que la tortue doit prendre. Comme sur une boussole :
le nord (haut de l'écran) correspond à un cap Ø
l'est (droite de l'écran) correspond à un cap 9Ø
le sud (bas de l'écran) correspond à un cap 18Ø
l'ouest (gauche de l'écran) correspond à un cap 27Ø
et toutes les positions intermédiaires.

L'instruction CAP donne le cap actuel de la tortue.

La position de la tortue dans le quadrillage est repérée par une liste de deux nombres.

Le centre de l'écran correspond à [Ø Ø]. Par ce centre passent deux droites du quadrillage : l'une "horizontale" (axe des x) l'autre "verticale" (axe des y). En dehors du centre le premier nombre indique la position horizontale et le second la position verticale.

Par exemple [1Ø 2Ø] est au croisement de la 10e verticale à droite du centre et de la 20e horizontale au-dessus du centre et [– 1Ø – 2Ø] est au croisement de la 10e verticale à gauche du centre et de la 20e horizontale au dessous du centre. Ceci correspond à un système de coordonnées dont l'origine est le centre de l'écran.

L'instruction FPOS suivie de la liste de deux nombres (écrits entre crochets) indique à la tortue le point de l'écran sur lequel elle doit se poser (1).
L'instruction POS donne les coordonnées du point sur lequel se trouve la tortue.

b) La forme du champ

— CLOS
Quand le champ est CLOS (dans le cas du démarrage, par exemple) la tortue est "cloîtrée" dans l'écran. Elle se déplace uniquement à l'intérieur de ces limites, et toute tentative de dépassement produit l'affichage du message Logo :

LA TORTUE VA SORTIR

— FEN pour Fenêtre

Sous ce mode, la tortue peut se déplacer dans un champ beaucoup plus grand que l'écran. Vous n'en voyez qu'une partie car votre écran est

---

(1) Il est possible d'indiquer plus d'un point par FPOS. La tortue se posera successivement sur ces points.

une fenêtre ouverte au milieu de cette grande étendue... Quand vous donnez une instruction qui risque de mettre la tortue en dehors de l'écran, elle part... Vous ne la voyez plus. Elle exécutera vos autres instructions hors de votre vue. Vous pouvez alors demander à la machine sa position (POS) et elle vous répondra. Vous pouvez aussi ordonner : ORIGINE ce qui correspond à FPOS [Ø Ø] FCAP Ø.

— ENR pour Enroule

La tortue sort de l'écran quand vous lui donnez une instruction qui dépasse les limites de l'écran. Elle réapparaît de l'autre côté sur la même horizontale ou verticale. La tortue disparaît et réapparaît ; si sa direction est oblique, elle trace des zébrures sur l'écran à chacun de ses passages.

c) La couleur du champ

Vous pouvez choisir vous-même la couleur que vous voulez donner au champ de la tortue. Le code des couleurs du champ est le même que celui du crayon.

La primitive FCFG suivie du code de la couleur fixe le champ à la couleur demandée. CF donne le code de la couleur actuelle du champ. Attention : si vous choisissez la même couleur pour le champ et pour le crayon vous ne verrez plus les traces de votre tortue !

3. L'écran graphique

En dessinant, vous avez peut-être déjà remarqué que certaines lignes obliques ressemblent à des rectangles reliés par la diagonale.
D'autre part, quand deux traits de couleur différente se croisent ou se rapprochent, vous pouvez avoir des modifications de couleur. En effet, sur une ligne horizontale, les repères de l'écran graphique correspondent à des segments de 8 points qui peuvent comporter simultanément 2 couleurs : une pour le fond, et une pour le tracé. Ainsi, quand une troisième couleur intervient, (un nouveau tracé) immédiatement tout le segment prend cette nouvelle couleur de tracé.

4. La tortue dans son champ

Il y a deux façons de guider la tortue.

— En la "pilotant" indépendamment de l'endroit précis où elle se trouve sur le moment.

— En indiquant le point de l'écran sur lequel elle doit se positionner par l'instruction FPOS, et l'orientation qu'elle doit prendre par l'instruction FCAP.

Voici l'exemple d'une procédure permettant de tracer un carré de 50 pas de côté.

```
POUR CARRE1
FPOS [0 50]
FPOS [50 50]
FPOS [50 0]
FPOS [0 0]
FIN
```

ou aussi

```
POUR CARRE2
FPOS [0 50 50 50 50 0 0 0]
FIN
```

Pour tracer ce carré, le cap ne joue aucun rôle.

Le carré a alors un emplacement bien défini sur l'écran. Ainsi, quelle que soit la position de la tortue, CARRE1 ou CARRE2 la ramènent au centre pour tracer le carré toujours au même endroit.

Nous connaissons déjà des primitives qui commandent des déplacements de la tortue : AVANCE, RECULE, TD, TG. Ces primitives considèrent la tortue indépendamment du lieu où elle se trouve et ne font référence qu'à son "corps" : AVANCE 80 lui fait suivre la direction de son "regard", d'un nombre de pas égal à 80.

Chapitre 4

Procédures avec données

Vous connaissez déjà les primitives utilisées dans la partie FAIRE de ce chapitre. Nous voulons attirer votre attention sur un de leurs aspects spécifiques : la donnée.

Faire

```
?AVANCE 3
```

La tortue avance de 3 pas.

```
?AVANCE 30
```

Elle avance encore de 30 pas.
Dans les deux cas, l'action effectuée est la même : avancer. Ce qui change, c'est la distance parcourue. Le nombre de pas est une information que l'on doit obligatoirement indiquer quand on ordonne AVANCE. C'est ce qu'on appelle une **donnée**. Nous avons déjà vu plusieurs primitives avec donnée.

Et les procédures ?

```
POUR CARREDIX
REPETE 4 [AVANCE 10 TD 90]
FIN


POUR CARRECINQUANTE
REPETE 4 [AVANCE 50 TD 90]
FIN

POUR CARRECENT
REPETE 4 [AVANCE 100 TD 90]
FIN
```

Demandons la réalisation de ces procédures :

```
?CARREDIX
```

Et voici un tout petit carré de 10 pas de côté.

```
?CARRECINQUANTE
```

Ce carré-ci est déjà plus visible.

```
?CARRECENT
```

Et celui-ci est bien grand.

Essayons autre chose

```
POUR SALUERACHILLE
ECRIS "BIENVENUE
ECRIS "ACHILLE
FIN

?SALUERACHILLE
BIENVENUE
ACHILLE

POUR SALUERALICE
ECRIS "BIENVENUE
ECRIS "ALICE
FIN

?SALUERALICE
BIENVENUE
ALICE
```

Comment définir de façon générale une procédure qui trace des carrés de toutes dimensions ou une procédure qui puisse saluer n'importe qui ? Il faudrait alors préciser respectivement le nombre de pas du côté (pour le carré) ou le nom de la personne à saluer quand on demande l'exécution de la procédure.

Apprendre à la machine comment faire

1. Tous les carrés sont semblables.

Pour faire un carré, il faut toujours avancer et tourner de 90 degrés quatre fois de suite. Ce qui peut varier d'un carré à un autre c'est le nombre de pas dont la tortue doit avancer c'est-à-dire le côté du carré. Voici comment écrire cette procédure en Logo :

```
POUR CARRE :COTE
REPETE 4 [AVANCE :COTE TD 90]
FIN
```

Pour taper : appuyez en même temps sur les touches [•] et [•] Ne laissez pas d'espace entre : et le mot COTE

En demandant l'exécution de CARRE il faudra préciser la longueur souhaitée du côté :

```
?CARRE 20
?CARRE 50
?CARRE 100
```

Et voici avec une seule procédure toute la famille des carrés !

2. Une suite de carrés

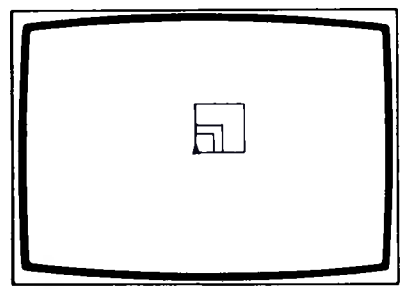
On peut alors utiliser plusieurs carrés de côtés différents pour dessiner une suite de carrés comme celle-ci :

```
POUR SUITE
CARRE 50
CARRE 30
CARRE 20
FIN
```

Essayons :

```
?SUITE
```

Et voilà... un... deux... et trois carrés !



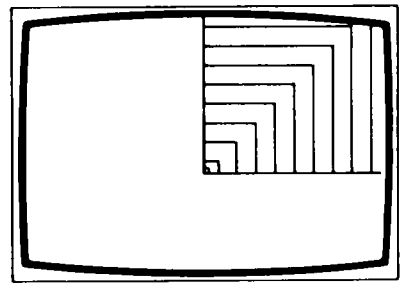
3. Les carrés s'agrandissent

Le dessin de carrés emboîtés n'est pas difficile à réaliser.
La tortue trace des carrés dont la longueur du côté augmente à chaque fois de 10 pas. Voici comment ceci peut s'écrire :

```
POUR IMAGE :COTE
CARRE :COTE
IMAGE :COTE + 10
FIN
```

Voici encore une procédure récursive. Mais, dans ce cas, chaque fois que la tortue "fait" IMAGE le carré devient plus grand de 10 pas... Essayons-la :

```
?IMAGE 3
```



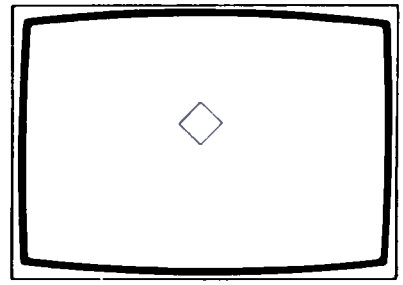
La tortue commence par tracer un carré de 3 pas de côté puis un carré de 13 pas de côté puis un carré de 23 pas...
Si le champ est CLOS, elle s'arrête après CARRE 93 et le message DANS CARRE, LA TORTUE VA SORTIR s'affiche.
Si le champ est en mode FENêtre ou ENRoule appuyez sur les touches [CNT] et [C] pour arrêter la procédure.

4. Un carré penché

A partir de CARRE :COTE, en penchant le carré un peu on pourrait en faire un CARREPenché.

```
POUR CARREP :COTE
TD 45
CARRE :COTE
CT
FIN
```

```
?CARREP 35
```

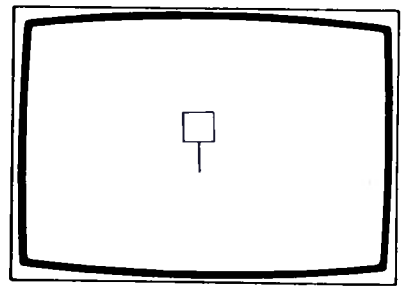


5. Un panneau carré

Puisque la tortue sait faire des carrés de tailles différentes, nous pouvons lui apprendre à faire des panneaux.

```
POUR PANNEAU :TAILLE
CARRE :TAILLE
TD 90 AVANCE :TAILLE / 2
TD 90 AVANCE :TAILLE
CT
FIN


?PANNEAU 30
?VE
?PANNEAU 50
```



Bon ! Le piquet est toujours au milieu et sa taille est proportionnelle à celle du carré.

6. Les carrés tournants

On peut maintenant réécrire les carrés tournants du Chapitre 3 en laissant au choix la taille du carré et l'angle de rotation.

```
POUR TOURNECARRE :GRANDEUR :ANGLE
FCC CC + 1
CARRE :GRANDEUR
TD :ANGLE
TOURNECARRE :GRANDEUR :ANGLE
FIN
```

Essayons cette procédure.

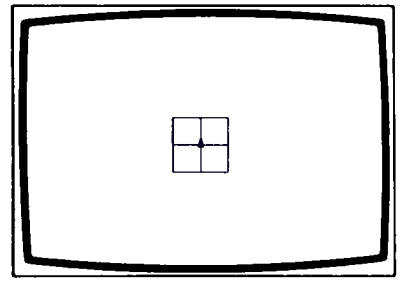
Si vous avez écrit :

```
?TOURNECARRE 30
PAS ASSEZ DE DONNEES POUR TOURNECARRE
```

Il faut préciser le côté du carré et l'angle au moment de la demande d'exécution.

Allons-y :

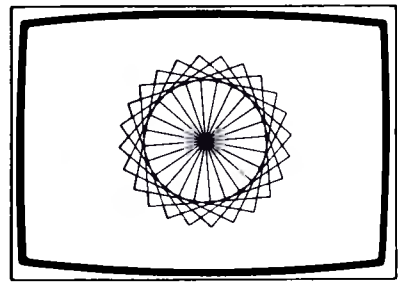
```
?TOURNECARRE 30 90
```



Ceci ressemble à 4 carrés. (CNT C pour arrêter la procédure).

En changeant simplement les nombres, on peut avoir :

```
?TOURNECARRE 40 20
```



Essayez-en d'autres...

7. Saluer tout le monde

Faisons la même chose pour les salutations :

```
POUR SALUER :NOM
ECRIS "BIENVENUE
ECRIS :NOM
FIN

?SALUER "MICHELE
BIENVENUE
MICHELE
```

8. ... Et la maison !

Nous pouvons maintenant faire des maisons petites et grandes avec la même procédure.

La procédure CARRE :COTE peut être utilisée pour faire aussi bien la façade que les fenêtres.

Faisons un toit qui puisse aller sur chaque maison.

```
POUR TRIANGLE :COTE
REPETE 3 [AVANCE :COTE TD 120]
FIN
```

Et voici la recette pour loger chacun à sa taille :

```
POUR MAISON :X
CT FCFG Ø FCC 7
CARRE :X
AVANCE :X TD 3Ø
FCC 1
TRIANGLE :X
TD 6Ø LC AVANCE 2 * ( :X / 5 )
TD 9Ø AVANCE :X / 5 BC
FCC 2
CARRE :X / 5 LC TG 9Ø
AVANCE :X / 5
BC CARRE :X / 5
FIN
```

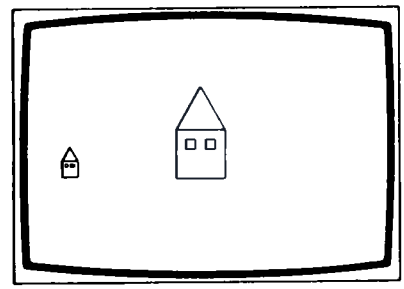
Avance de deux fois le contenu de : x/5

```
?MAISON 5Ø
```

Et voici la maison d'un OGRE !

```
?LC FPOS [-9Ø Ø] FCAP Ø BC MAISON 15
```

Voici la maison de PLOUM !



A suivre...

Penser à ce qu'on a fait

1. La donnée

Une donnée est une information (nombre, mot, liste) qui accompagne obligatoirement certaines primitives ou procédures. Par exemple AVANCE ou ECRIS sont des primitives qui ont besoin d'une donnée. Quand on les utilise on doit spécifier AVANCE "combien" ou ECRIS

"quoi" à la machine. D'autres primitives comme MT ou VE n'ont pas besoin de donnée.

Il en est de même pour les procédures. Nous pouvons définir des procédures avec donnée. Pour que ce soit une procédure avec donnée, celle-ci doit apparaître dans le titre de la procédure. CARRE :COTE (exemple 1) est une procédure avec donnée ; il faut préciser à la machine CARRE "combien" quand on lui demande de l'exécuter. Par contre, SUITE (exemple 2) n'est pas une procédure avec donnée : les informations relatives au carré sont fixées une fois pour toutes à l'intérieur de la procédure.

2. Procédures avec données

a) Écriture

Dans le titre de la procédure on représente la donnée par une suite de caractères (mot) précédée de deux points. S'il y a plusieures données, on les écrit les unes à la suite des autres, séparées par des espaces.

Par exemple :

```
POUR CARRE :COTE
POUR TOURNECARRE :GRANDEUR :ANGLE
```

Il peut être plus parlant de choisir un mot qui exprime ce que représente la donnée. :COTE symbolise la longueur du côté du carré.

Dans le corps de la procédure le ou les noms respectifs de ces données doivent figurer aux places où leurs valeurs respectives seront utilisées :

Par exemple :

```
POUR RECTANGLE :LONGUEUR :LARGEUR
AVANCE :LONGUEUR TD 90
AVANCE :LARGEUR TD 90
AVANCE :LONGUEUR TD 90
AVANCE :LARGEUR TD 90
FIN
```

LONGUEUR et LARGEUR sont les noms des données respectives qui seront utilisées pour construire le RECTANGLE.

b) Exécution

Au moment de la demande d'exécution d'une procédure avec donnée, il faut attribuer un **objet Logo,** c'est-à-dire un mot, un nombre ou une liste à chaque nom de donnée (c'est ce qu'on appelle aussi une "chose" ou une valeur).

Reprenons la procédure CARRE :COTE. On peut se représenter :COTE comme une boîte étiquetée COTE qui pour le moment est vide. Lors de la demande d'exécution de la procédure par exemple CARRE 50, la boîte étiquetée COTE reçoit le nombre 50 et pendant l'exécution à chaque fois que la machine rencontrera :COTE elle regardera ce qui se trouve dans la boîte ici, 50, et la réalisation de CARRE 50 est un carré de 50 pas de côté.
Dès la fin de l'exécution, la boîte étiquetée COTE redevient vide.
Pour cette raison si nous demandons :

```
?CARRE
PAS ASSEZ DE DONNEES POUR CARRE
```

car l'ordinateur ne trouve rien dans la boîte étiquetée COTE.
Lorsque la procédure dépend de plusieurs données, il faut attribuer un objet Logo à chaque nom de donnée dans l'ordre de leur écriture.

Par exemple, lors de l'exécution :

```
RECTANGLE 100 20
```

La boîte étiquetée LONGUEUR recevra l'objet Logo 100 et celle étiquetée LARGEUR l'objet Logo 20.

3. Procédures imbriquées

Reprenons les procédures CARRE :COTE et PANNEAU :TAILLE

```
POUR CARRE :COTE
REPETE 4 [AVANCE :COTE TD 90]
FIN


POUR PANNEAU :TAILLE
CARRE :TAILLE
TD 90 AVANCE :TAILLE / 2
TD 90 AVANCE :TAILLE
CT
FIN
```

Demandons :

```
?PANNEAU 50
```

Et la boîte TAILLE reçoit la valeur 50.
PANNEAU 50, au début de l'exécution, "appelle" CARRE 50. La boîte COTE (nom choisi pour la donnée lors de la définition de CARRE) reçoit la valeur 50 (trouvée dans la boîte TAILLE).

L'ordinateur retient la quantité de données dont dépend une procédure et les places respectives occupées par ces données. Ainsi, une même procédure peut être utilisée avec des noms de données différents. C'est le cas ici pour la procédure CARRE dont le nom de la donnée peut être COTE ou TAILLE ou X (voir MAISON)...

4. Opérations arithmétiques sur les données

Il est possible d'écrire "formellement" toutes les opérations arithmétiques sur les noms des données.
Par exemple dans PANNEAU :TAILLE, :TAILLE / 2 prendra au moment de l'exécution la moitié de la valeur attribuée à TAILLE.
Tous les symboles arithmétiques peuvent ainsi être utilisés pour combiner des noms de données à condition, évidemment, que lors de l'exécution ces données soient des nombres.

Chapitre 5

De beaux dessins...

Ce chapitre est consacré à l'écriture de procédures dont la réalisation graphique est agréable à regarder.

Il ne s'agira donc pas de FAIRE mais :

Apprendre à la machine comment faire

1. Les dessins géométriques

a) La famille des polygones

Un polygone régulier est une figure géométrique dont les côtés ont la même longueur et dont les angles sont égaux.

a)

```
POUR POLY :COTE :ANGLE
AVANCE :COTE TD :ANGLE
POLY :COTE :ANGLE
FIN
```

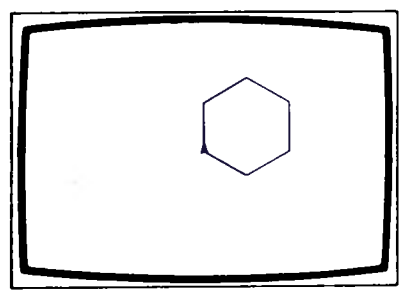
POLY est une procédure récursive dont la réalisation est un "mouvement perpétuel"... CNT C permet de l'interrompre !

```
?POLY 50 90
```

C'est un carré

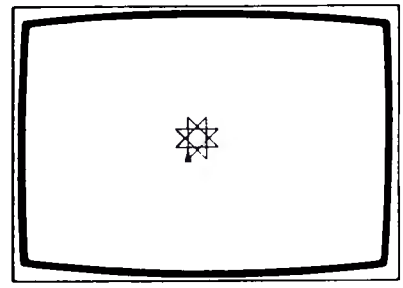
```
?POLY 50 60
```

Un hexagone



?POLY 50 135

Une étoile



b) La famille des polygones : triangle, carré, pentagone, hexagone etc. peut également être construite par une procédure qui n'est pas récursive.
En effet si nous regardons la définition
du triangle :

```
REPETE 3 [AV :COTE TD 120]
```

où 120 = 360/3 degrés

du carré :

```
REPETE 4[AV :COTE TD 90]
```

où 90 = 360/4 degrés

ou du pentagone :

```
REPETE 5 [AV :COTE TD 72]
```

où 72 = 360/5 degrés

Ceci peut se généraliser en :

```
POUR POLY1 :N :COTE                    :N contiendra le nombre de côtés du polygone
REPETE :N [AVANCE :COTE TD 360 / :N]   :COTE contiendra la longueur
FIN                                    de chaque côté.
```

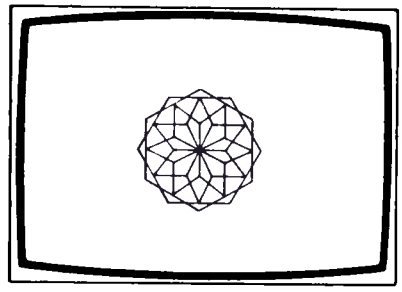
POLY1 360 0.5 est une approximation du cercle.

c) A partir de cette procédure qui s'arrête, nous pouvons dessiner un polygone qui tourne et se reconstruit à chaque fois

```
POUR TPOLY1 :N :COTE :ROT      :ROT contiendra l'angle de rotation
POLY1 :N :COTE
TD :ROT
TPOLY1 :N :COTE :ROT
FIN
```

:N :COTE :ROT sont les données de TPOLY1

```
?TPOLY1 6 25 30
```



d) Si vous introduisez la couleur dans vos dessins, ils seront invisibles si le crayon et le fond ont la même couleur. Essayons d'éliminer la couleur du crayon qui est identique à celle du fond.
La primitive qui nous donne la couleur du fond est CF.

```
?ECRIS CF
```

La primitive EGAL? suivie de deux nombres (ou mots, ou listes) vérifie s'ils sont identiques.
Écrivons la procédure : (1)

```
POUR ELIMINE
SI EGAL? CC CF [FCC CC + 1]
FIN
```

Voici ce que dit cette procédure en français :
Si la couleur du fond est la même que celle du crayon (SI EGAL? CC CF) alors, fixe la couleur du crayon (FCC), à la couleur suivante (CC + 1).

ELIMINE appelé avec toutes les procédures qui utilisent la couleur évitera les écritures invisibles !

---

(1) La primitive SI introduit une **expression conditionnelle**.
SI <telle condition est vraie> alors <fais telle action> sinon passe à l'action suivante.

Par exemple :

```
?FCC 5 FCFG 5 AVANCE 50
```

La tortue a avancé de 50 pas mais en traçant en violet sur fond violet. Nous ne voyons rien !

```
?ELIMINE AVANCE 50
```

Cette fois la couleur du crayon est bleu clair ce qui correspond au code 6 et la tortue a laissé une trace visible.

b) La famille des spirales

Reprenons la procédure de l'exemple a) qui définit les polygones de façon récursive. Nous allons écrire une procédure analogue telle qu'à chaque appel :

— soit la longueur du côté augmente

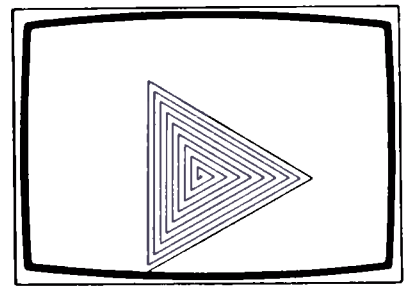
— soit la mesure de l'angle augmente

a) Incrémenter la donnée :COTE

```
POUR SPI :COTE :ANGLE
AVANCE :COTE TD :ANGLE
SPI :COTE + 5 :ANGLE
FIN
```

```
?SPI 0 120
```

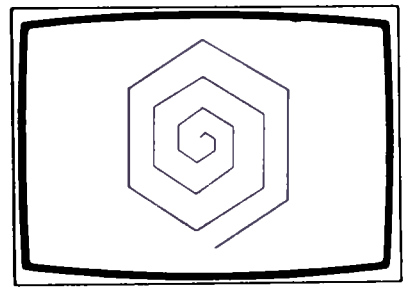
Si le champ est clos la réalisation de SPI sera limitée par les bords de l'écran.



C'est une spirale construite à partir d'un triangle.

A partir d'un hexagone on obtient :

```
?SPI Ø 6Ø
```



Vous pouvez faire arrêter la procédure si la longueur du côté dépasse 8Ø :

```
POUR SPI1 :COTE :ANGLE
SI :COTE > 8Ø [STOP]
AVANCE :COTE TD :ANGLE
SPI1 :COTE + 5 :ANGLE
FIN
```

Dès que le contenu de :COTE est supérieur à 80 la procédure s'arrête sinon elle passe à la ligne suivante. (1)

Vous pouvez colorer ces effets, changer l'incrémentation de la longueur du côté etc.

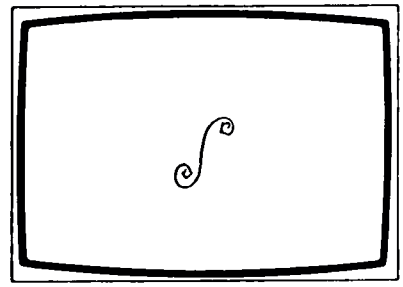
b) Incrémenter la donnée :ANGLE. Les primitives TD et TG n'acceptent pas des données supérieures à 360 dans la procédure suivante, dès que l'angle dépasse 360 degrés, il est ramené à un angle équivalent mais de mesure inférieure à 360. Par exemple, 363 degrés et 3 degrés représentent le même angle.

```
POUR ESPI :COTE :ANGLE
SI :ANGLE > 36Ø [ESPI :COTE :ANGLE - 36Ø STOP]
AVANCE :COTE TD :ANGLE
ESPI :COTE :ANGLE + 1Ø
FIN
```

---

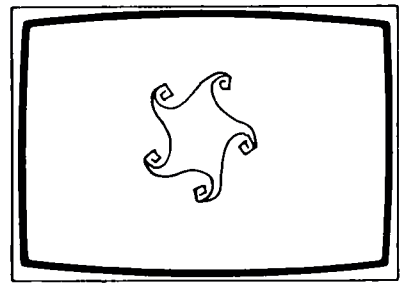
(1) > compare les deux nombres qui l'entourent. L'expression est vraie quand le nombre de gauche est plus grand que celui de droite.

?ESPI 10 0



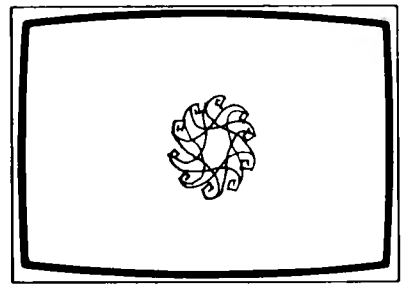
Un S sur lequel la tortue repasse indéfiniment !

?ESPI 10 3



Une ronde de S

?ESPI 10 2



Une autre ronde.

2. Dessins "naïfs" à partir d'arcs de cercles

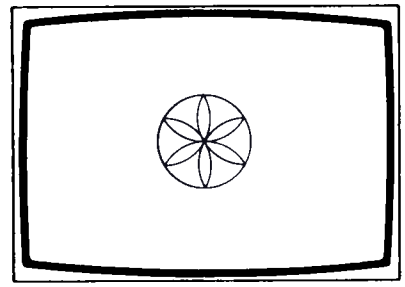
Nous allons définir des arcs dont nous pourrons faire varier aussi bien la taille que la courbure.
Un arc obtenu en tournant toujours à droite :

```
POUR ARCD :N :PAS
REPETE :N [AVANCE :PAS TD 1]
FIN
```

Celui obtenu en tournant toujours à gauche :

```
POUR ARCG :N :PAS
REPETE :N [AVANCE :PAS TG 1]
FIN
```

Les rosaces :



```
POUR ROSACE
CT
ARCD 360 1
TD 60
REPETE 3 [ARCD 120 1 TD 120]
TG 60 ARCD 60 1 TD 60
REPETE 3 [ARCD 120 1 TD 120]
FIN
```

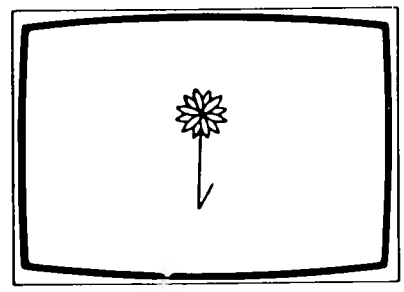
Les fleurs :
Parcourir un pétale nous fait tourner de 360 degrés.
Si chaque arc de cercle tourne de :N degrés il faudra à chaque extrémité de pétale tourner de 180 − :N degrés:

```
POUR PETALE :N :PAS
ARCD :N :PAS
TD 180 - :N
ARCD :N :PAS
FIN

?PETALE 30 1
```

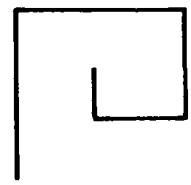
De là à faire une marguerite – il n'y a qu'une procédure.

```
POUR MARGUERITE
CT FCC 3 REPETE 12 [PETALE 30 0.5]
RE 20 FCC 2 RECULE 20
TD 20 PETALE 10 1
FIN
```



3. Les frises

Construisons une frise en reproduisant plusieurs fois le motif suivant :



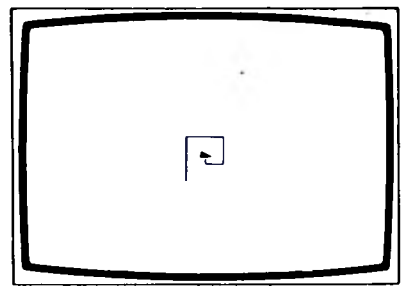
On peut l'obtenir en tapant :

```
AV 50 TD 90 AV 40 TD 90 AV 30 TD 90 AV 20 TD 90 AV 10
```

ce qui peut être défini par une procédure permettant de choisir la grandeur du motif :

```
POUR GRECQUE1 :X
SI :X < 5 [STOP]
AVANCE :X TD 90
GRECQUE1 :X - 10
FIN
```

```
?GRECQUE1 50
```



Cependant la tortue n'est pas revenue à son point de départ. Pour qu'elle puisse faire le chemin inverse, à chaque fois que la tortue fait AV :X TD 90 il faut lui faire faire TG 90 RE :X.

Modifions GRECQUE de la manière suivante :

```
POUR GRECQUE :X
SI :X < 5 [STOP]
AVANCE :X TD 90
GRECQUE :X - 10
TG 90 RECULE :X
FIN
```

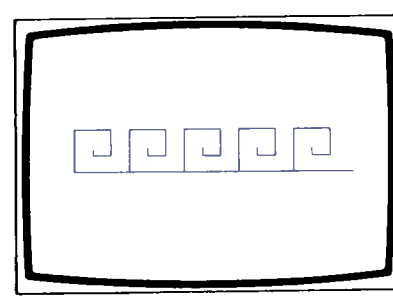
```
?GRECQUE 50
```

C'est que nous voulions : après avoir dessiné le motif initial, la tortue l'a reparcouru en "arrière". GRECQUE est une procédure récursive, mais dans laquelle il reste des actions à accomplir après l'appel récursif. C'est ce qu'on appelle une **récursivité non terminale.**
et la frise :

```
POUR FRISE
LC FPOS [-150 0]
BC REPETE 5 [GRECQUE 50 TD 90 AVANCE 60 TG 90]
FIN
```

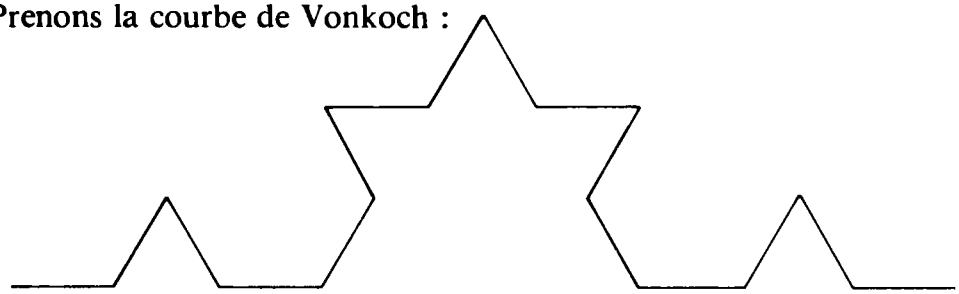
Vous pouvez bien sûr faire vos frises en couleur surtout pour les couleurs négatives de crayon.

```
?VE CT FCC -2   FRISE
```



4. Un flocon de neige

Logo permet de réaliser facilement de jolies courbes appelées fractales. Prenons la courbe de Vonkoch :



Voici la procédure qui permet de la réaliser en Logo :

```
POUR VONKOCH :X
SI :X < 5 [AVANCE :X STOP]
VONKOCH :X / 3 TG 60
VONKOCH :X / 3 TD 120
VONKOCH :X / 3 TG 60
VONKOCH :X / 3
FIN
```

Elle peut créer un flocon de neige :

```
POUR FLOCON :X
REPETE 3 [VONKOCH :X TD 120]
FIN

?FLOCON 60
```

5. ... Et la maison !

Nous pouvons à partir de MAISON :X dessiner une rue en perspective :

```
POUR PERSP :X
SI :X < 10 [STOP]                          Si le coté est inférieur à 10 STOP
BC MAISON :X                               Maison
LC AVANCE 2 * ( :X / 5 ) FCAP 0            Position pour tracer la prochaine
PERSP :X - 10                              Appel récursif
FIN

?ME 1 LC FPOS [-120 -90] BC PERSP 60
```

Une jolie perspective

```
POUR PERSPECTIVE
ME 1 FCFG 8
LC FPOS [-120 -90]
BC PERSP 60
FIN
```

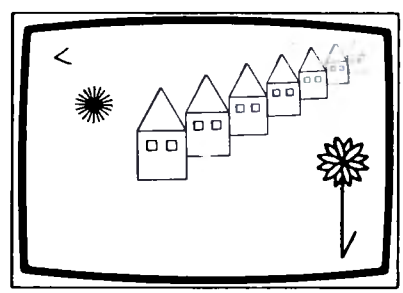
Embellissons : une fleur, un oiseau... On se croirait dans un village !...

```
POUR RUE
ENR
PERSPECTIVE LC FPOS [80 -50] BC
MARGUERITE
LC FPOS [-80 60] FCC 3 BC
REPETE 20 [AVANCE 20 RECULE 20 TD 18]
LC VOLER
FIN
```

n'oubliez pas de "charger" la procédure VOLER et toutes celles dont elle dépend

A suivre...

```
?RUE
```



Penser à ce qu'on a fait

1. Les tests d'arrêt

Les tests d'arrêt sont des expressions conditionnelles qui sont libellés ainsi :

```
SI... [STOP]
```

Par exemple :

```
SI :COTE > 80 [STOP]
```

Cette phrase Logo signifie que dès que la condition introduite par SI... est vraie (dans ce cas, la valeur :COTE supérieure à 80), la procédure s'arrête.

Faites attention à la ligne à laquelle vous introduisez le test d'arrêt car ceci a des conséquences pour le déroulement de la procédure.

2. La récursivité non terminale

Reprenons la procédure GRECQUE

```
POUR GRECQUE :X
SI :X < 5 [STOP]
AVANCE :X TD 90
GRECQUE :X - 10
TG 90 RECULE :X
FIN
```

Remarquez que l'appel récursif ne se situe pas à la dernière ligne comme c'était le cas jusqu'à présent.

Essayons de comprendre ce qui se passe quand nous demandons l'exécution de :

```
GRECQUE 30
```

La "chaîne" des GRECQUE s'arrête à GRECQUE 0 et la procédure se déroule de la manière suivante :

GRECQUE 30 appelle GRECQUE 20 qui appelle GRECQUE 10 qui appelle GRECQUE 0 qui appelle STOP.

Chaque GRECQUE fait avancer et tourner la tortue, appelle une GRECQUE plus petite, puis tourne à gauche et recule ; cette dernière action ne peut se réaliser que lorsque la GRECQUE plus petite a été dessinée.

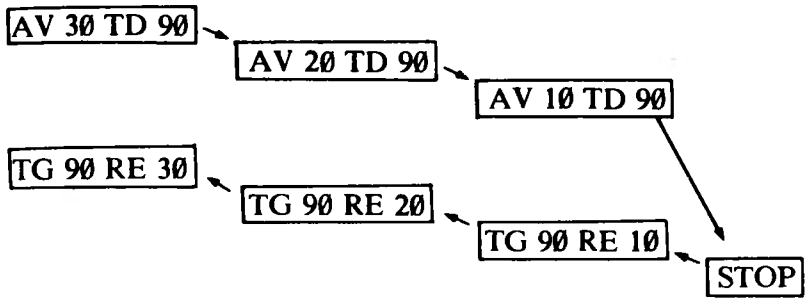
Jusqu'à l'appel de GRECQUE 0 les actions effectuées ont été :

AV 30 TD 90
AV 20 TD 90
AV 10 TD 90
STOP

Le STOP permet alors à GRECQUE 10 de terminer son dessin, puis à GRECQUE 20, puis à GRECQUE 30, c'est-à-dire : (de haut en bas)

TG 90 RE 10
TG 90 RE 20
TG 90 RE 30

Si nous enchaînons ces actions, la procédure GRECQUE 3Ø se déroule selon le parcours suivant :



Chapitre 6

Des mots, des noms et des instructions... définitions et écriture

Dialogue

— Savez-vous écrire "PLOUM" ? Évidemment il faut un P puis un L puis un O puis un U puis un M

— Mais qu'est-ce que "PLOUM" ?

— C'est mon chien, regardez : PLOUM ! Vous voyez il accourt...

Voici donc PLOUM écrit trois fois et à chaque fois son sens est différent.

La première fois "PLOUM" est un mot ou encore un ensemble de caractères écrits dans un ordre précis (chaîne de caractères).
La deuxième fois PLOUM est le nom d'un chien.
La troisième fois PLOUM est l'ordre donné au chien de venir immédiatement auprès de son maître.

En français, ces différentes significations pour un même mot se comprennent soit par l'intonation de la voix, dans le langage parlé, soit par l'utilisation de symboles dans le langage écrit. Par exemple deux points (:) précèdent une explication, des guillemets (") entourent l'écriture d'un nom, un point d'exclamation (!) intime un ordre.

Quels symboles utilise-t-on en Logo ?

— un ordre n'est annoncé par aucun symbole : entouré d'espaces c'est une action à exécuter.

— un guillemet (") annonce un mot (une chaîne de caractères). Par exemple, "PLOUM

— deux points (:) annoncent le nom d'une "chose" par exemple, :PLOUM

Essayons donc en Logo :

```
?ECRIS "PLOUM
PLOUM
```

"PLOUM représente la chaîne de caractères dont on demande l'affichage.

```
?ECRIS :PLOUM
PAS DE CHOSE DONNE A PLOUM
```

Effectivement, c'est dans le langage courant que PLOUM désignait un chien. Comment apprendre à l'ordinateur que PLOUM est le nom d'un chien ? En lui disant justement que la suite de caractères P L O U M est le nom attribué au mot chien.
Ce qui s'écrit en Logo :

```
?DONNE "PLOUM "CHIEN
```

DONNE est la primitive qui permet d'affecter un nom à une "chose". Voyons donc ce qui est désigné par le nom PLOUM

```
?ECRIS :PLOUM
CHIEN
```

CHIEN est une information un peu brève, si nous voulons que le nom de PLOUM désigne : **un petit caniche noir** il faut considérer la phrase UN PETIT CANICHE NOIR comme une liste de mots.

En Logo une liste est délimitée par des crochets.
Nous avons déjà rencontré cette notation aux chapitres précédents pour l'écriture de phrase ECRIS [JE VOUS SALUE], l'exécution d'une série d'ordres REPETE 4 [AV 5Ø TD 9Ø], ou les coordonnées d'un point FPOS [15 45] etc.

Maintenant nous pouvons donner une autre signification au mot PLOUM

```
?DONNE "PLOUM [UN PETIT CANICHE NOIR]
?ECRIS :PLOUM
UN PETIT CANICHE NOIR
```

La définition précédente est oubliée.

... Et la maison !

Enfin notre maison aura une adresse !

```
?DONNE "MAISON [2 RUE DE LA TORTUE]

POUR ADRESSE
MAISON 5Ø
ECRIS :MAISON
FIN

?ADRESSE
```

n'oubliez pas d'avoir la procédure MAISON et toutes celles dont elle dépend en mémoire

A suivre...

Penser à ce qu'on a fait

1. Les mots, les noms, les listes et les ordres

Lorsque l'ordinateur lit de gauche à droite une suite d'instructions à effectuer, s'il rencontre :

aucun signe particulier il sait que ce qui suit est un ordre qu'il doit exécuter.

" il sait que ce qui suit jusqu'au prochain espace est un mot dont il n'a pas à connaître la signification.

[il sait que ce qui suit est une liste délimitée à droite par le dernier crochet fermant]

: il sait ce qui suit est le nom d'une "chose", cette chose étant un nombre, un mot ou une liste.

Les nombres sont des mots particuliers. Suivant l'instruction dans laquelle ils interviennent, ils seront considérés soit comme information numérique soit comme mot même s'il ne sont pas précédés de guillemet. par exemple :

```
?ECRIS 5
5
```

2. Les noms et les procédures

Les noms sont les mots qui désignent une chose (objet Logo)
Il y a deux sortes de noms :

1) les noms qui désignent une chose indépendemment de toute procédure.

a) Donner un nom à un objet Logo peut se faire en mode direct :

```
?DONNE "CHIEN "PLOUM
```

:CHIEN est alors le mot PLOUM, et si on demande

```
?ECRIS :CHIEN
PLOUM
```

Il est toujours possible de changer la chose désignée par le nom CHIEN, toujours par un DONNE

```
?DONNE "CHIEN "MEDOR
```

:CHIEN est alors MEDOR

b) Donner un nom à un objet Logo peut aussi se faire dans une procédure mais à condition que ce nom ne figure pas dans le titre de la procédure en tant que nom de donnée (nous verrons plus loin pourquoi).

```
POUR CARRE :COTE
DONNE "DESSIN [CARRE]
REPETE 4 [AVANCE :COTE TD 90]
FIN
```

Avant la première exécution de ce CARRE, :DESSIN n'a pas reçu de valeur mais dès que ce CARRE aura été exécuté, DESSIN sera le nom de la chose [CARRE].

2) Les noms qui apparaissent dans les titres des procédures :
Prenons un exemple :

```
POUR CARRE :COTE
ECRIS :COTE
REPETE 4 [AVANCE :COTE TD 90]
ECRIS :COTE
FIN
```

COTE doit obligatoirement recevoir une chose (objet Logo, dans ce cas, un nombre) lors de la demande d'exécution de la procédure.
Par exemple l'exécution de CARRE 50 donnera la chose 50 au mot CARRE et CARRE 50 fera afficher 50, puis tracera un carré de 50 pas de côté puis affichera à nouveau 50. A la fin de la procédure :COTE reprend sa valeur initiale, c'est-à-dire :

— si précédemment COTE a été le nom donné à une chose, par la primitive DONNE, alors COTE redevient le nom de cette chose

— sinon COTE n'est plus un nom, c'est-à-dire ne dénomme rien du tout. Par exemple, dans ce second cas :

```
?ECRIS :COTE   provoquera l'affichage du message
PAS DE CHOSE DONNEE A COTE
```

Par contre si avant de définir CARRE nous avions donné la chose 100 à COTE par un DONNE "COTE 100, après l'exécution de CARRE 50 dont le déroulement est inchangé, si nous demandons

```
?ECRIS  :COTE
```

100 s'affichera

Supposons maintenant que COTE n'était pas le nom d'une chose, que se passerait-il si nous affections la chose 100 à COTE pendant le déroulement de la procédure CARRE ?

```
POUR CARRE :COTE
ECRIS :COTE
DONNE "COTE 100
ECRIS :COTE
REPETE 4 [AVANCE :COTE TD 90]
ECRIS :COTE
FIN
```

Faisons exécuter cette procédure

```
?CARRE 50
50
100
```
la tortue dessine le carré
```
100
```

COTE est lui le nom de la donnée 50
le DONNE a changé le contenu de COTE
100 est toujours la chose de COTE

Et maintenant que reste-t-il dans COTE ?

```
?ECRIS :COTE
PAS DE CHOSE DONNEE A COTE
```

COTE a toujours été considéré comme le nom de la donnée dont dépend CARRE et par conséquent on perd toutes les manipulations qui ont pu être faites sur ce nom à l'exécution de la procédure.

Chapitre 7

Chiffres et lettres

Dans ce chapitre, nous aborderons les opérations qui permettent de traiter les nombres (l'arithmétique), et celles qui permettent de traiter les listes et les mots.

Faire

1. Des chiffres

```
?ECRIS 5 + 7
12
```

ne pas oublier l'espace entre + et les nombres et entre ECRIS et la suite.

D'accord

```
?ECRIS SOMME 5 7
12
```

L'ordre SOMME produit le même calcul entre 5 et 7 que +.
Et si on ne demande pas l'écriture du résultat :

```
?5 + 7
QUE FAIRE DE 12
```

```
?SOMME 5 7
QUE FAIRE DE 12
```

L'opération a été effectuée mais on n'a donné aucune instruction d'utilisation de son résultat.

```
?AVANCE SOMME 40 60
```

La tortue avance de 100 pas (100 = 40 + 60)

```
?TD 45 + 45
```

La tortue pivote d'un quart de tour à droite.

```
?ECRIS SOMME 2 3 6
5
QUE FAIRE DE 6
```

Somme ne peut additionner que 2 nombres.

```
?ECRIS 8 - 5 (1)
3
```

(1) Laissez toujours un espace blanc entre le signe et les chiffres. Si vous écrivez −5, ceci indique le nombre négatif et non pas l'opération de soustraction.

```
?ECRIS PROD 3 6
18

?ECRIS 3 * 6
18
```

Comme précédemment l'ordre PROD effectue le produit des deux nombres qui le suivent et * effectue le produit des nombres qui l'entourent.

```
?ECRIS QUOT 30  8
3
```

quand on divise 30 par 8 le quotient est 3 et le reste est 6

Le premier nombre est donc le dividende et le second, le diviseur. Quel est alors le reste ?

```
?ECRIS RESTE 5 3
2
```

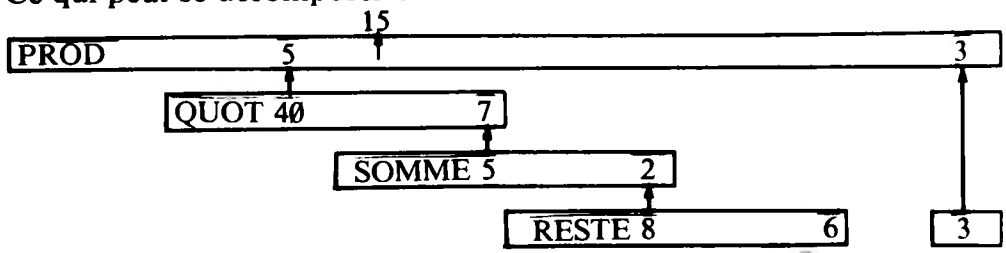
Si on oublie une donnée :

```
?ECRIS QUOT 5
PAS ASSEZ DE DONNEES POUR QUOT
```

Le message est clair !
Combinons tous ces ordres.

```
?ECRIS PROD QUOT 40 SOMME 5 RESTE 8 6 3
15
```

Ce qui peut se décomposer en :



Bien souvent le symbole de la division est /. Que représente-t-il en Logo ?

```
?ECRIS 10 / 50
.2

?ECRIS 8 / 3
2.666667

?ECRIS DIV 8 3
2.666667
```

DIV et / calculent le résultat décimal de la division du chiffre de gauche par celui de droite. 7 chiffres au maximum sont affichés et le dernier est arrondi.

Et si nous donnons un nombre de plus de 7 chiffres :

```
?ECRIS 123456789123456789
1.234567E17
```

Les 7 premiers chiffres affichés sont exacts. E17 indique un exposant 17, c'est-à-dire que la virgule doit être déplacée de 17 chiffres vers la droite.

2. Et les lettres

Les instructions précédentes décrivent les opérations fondamentales de l'arithmétique. Quelles opérations peuvent être effectuées sur les mots ou les listes ?

— La **concaténation** : c'est-à-dire la réunion de deux mots ou de deux listes.
Pour les mots, la primitive est MOT

```
?ECRIS MOT "POIRE "AU
POIREAU
```

Pour les listes la primitive est PH pour PHrase

```
?ECRIS PH [TOUT VA] [TRES BIEN]
TOUT VA TRES BIEN
```

Là aussi si nous demandons :

```
?MOT "VELO "CYCLISTE
QUE FAIRE DE VELOCYCLISTE
```

La réunion des deux mots a été effectuée mais le résultat n'a pas été utilisé comme donnée d'une procédure.

Combinons ces ordres

```
?EC PH MOT "HE "RCULE MOT "POIRE "AU
HERCULE POIREAU
```

— La **troncation** d'un mot ou d'une liste :

```
?ECRIS SP "RAVIS
AVIS

?ECRIS SP [TOUT VA BIEN]
VA BIEN
```

SP construit un mot à partir du mot donné Sans son Premier caractère, et une liste à partir d'une liste donnée Sans son Premier membre (qui peut être un mot ou une liste).

Pour obtenir un élément d'un mot ou d'une liste.

```
?ECRIS PREM "TOUT
T

?ECRIS PREM [TOUT VA BIEN]
TOUT
```

PREM extrait le premier caractère d'un mot considéré lui-même comme mot ou bien le premier élément d'une liste qui peut lui-même être un mot comme ci-dessus ou une liste.

Enfin une primitive VIDE? teste si un mot ou une liste est vide (c'est-à-dire ne contient respectivement aucun caractère).

Le mot vide est symbolisé par un guillemet suivi d'un blanc " ; la liste vide par [].

```
?ECRIS VIDE? "BONJOUR
FAUX

?ECRIS VIDE? []
VRAI
```

Combinons tous ces ordres

```
?ECRIS PH PREM [[PIERRE EST UN] GARCON] MOT "VELO "CYCLISTE
PIERRE EST UN VELOCYCLISTE
```

Il existe aussi les primitives DER et SD

DER d'un mot, ou d'une liste est le dernier élément du mot ou le dernier caractère d'une liste.

SD d'un mot est le mot sans son dernier caractère. SD d'une liste est la liste sans son dernier membre.

Apprendre à la machine comment faire (1)

1. Écrire une table de multiplication

```
POUR TABSIX :N
SI :N > 10 [STOP]
ECRIS PH PH PH [6 *] :N "= 6 * :N
TABSIX :N + 1
FIN
```

:N recevra le nombre de départ
après 10 on s'arrête
passage à la multiplication suivante

TABSIX 1 donne l'affichage de la "table de 6" et si on ne veut pas avoir à écrire 1.

```
POUR TABLESIX
TABSIX 1
FIN
```

```
?TABLESIX
6 * 1 = 6
...
6 * 9 = 54
6 * 10 = 60
```

2. Dessin au hasard

La primitive HASARD rend un nombre entier au hasard dans des limites précisées :

HASARD 50 est un nombre entier choisi au hasard entre 0 et 49. Regardons un tracé produit au hasard.

```
POUR DESSINHASARD
FCAP HASARD 360
FCC HASARD 8
AVANCE HASARD 50
DESSINHASARD
FIN
```

HASARD 360 = nombre compris entre 0 et 359

Impossible de vous montrer ce que vous obtiendrez !

---

(1) Un point d'exclamation à la fin de ligne d'une procédure indique que la ligne Logo continue sur la ligne suivante.

3. Au hasard des phrases

Il s'agit de construire des phrases au hasard à partir d'une liste de sujets, une liste de verbes et une liste de compléments. Fixons chacune de ces listes.

```
?DONNE "SUJET [[LA SOURIS] [L'ORDINATEUR] [PAULINE]
[BALTHAZAR]]
?DONNE "VERBE [MANGE EXECUTE AIME JOUE]
?DONNE " COMPL [[LE FROMAGE] [LES ORDRES] [LE SOLEIL] [AU
HASARD]]
```

La primitive ITEM permet de choisir un membre dans une liste. Par exemple ITEM 2 :SUJET est [L'ORDINATEUR] nous l'utiliserons pour construire des phrases.

```
POUR FRASASAR
ECRIS PH PH ITEM 1 + HASARD 4 :SUJET IT!
EM 1 + HASARD 4 :VERBE ITEM 1 + HASARD !
4 :COMPL
SI LISCAR = "O [FRASASAR] [STOP]
FIN
```

Pour éviter la valeur Ø de HASARD on écrit

```
ITEM 1 + HASARD 4
```

LISCAR attend une lettre tapée au clavier (voir Chapitre 8). Ici si cette lettre est O une autre phrase s'affiche à l'écran sinon la procédure s'arrête.

Essayons-la :

```
?FRASASAR
LA SOURIS JOUE LE FROMAGE
```

tapons O

```
L'ORDINATEUR AIME AU HASARD
```

tapons O

```
PAULINE MANGE LE SOLEIL
```

tapons O

```
BALTHAZAR EXECUTE LES ORDRES
```

tapons N

A vous de trouver des phrases plus amusantes, d'écrire des vers etc.

4. Affichage d'un mot ou d'une liste

```
POUR DICTE :QCC                  :QCC pouvant contenir un mot ou une liste
SI VIDE? :QCC [STOP]             test d'arrêt lorsque la donnée :QCC est vide
ECRIS PREM :QCC                  Affichage de la première lettre
DICTE SP :QCC                    appel récursif de la même procédure en ôtant son
FIN                              premier élément à :QCC.
```

```
?DICTE "SALUT
S
A
L
U
T
```

ou

```
?DICTE [TOUT VA BIEN]
TOUT
VA
BIEN
```

```
POUR AFFICHE :QCC
SI VIDE? :QCC [STOP]
ECRIS :QCC
AFFICHE SP :QCC
FIN
```

```
?AFFICHE "SALUT
SALUT
ALUT
LUT
UT
T
```

```
POUR ECRISENVERS :QCC
SI VIDE? :QCC [STOP]
ECRIS DER :QCC
ECRISENVERS SD :QCC
FIN
```

Exemple : l'écriture à l'envers (en commençant par la droite).

```
?ECRISENVERS "NOM
M
O
N
```

5. ... Et la maison !

Reprenons la procédure PERSPECTIVE du Chapitre 5 (Pour la réaliser il faut aussi les procédures MAISON, PERSP etc. en mémoire centrale).

Attribuons un propriétaire pour chaque maison dans l'ordre où elles se dessinent :

```
?DONNE "PROPRIO [[JEAN ET JEANINE] [SOPHIE] [PIERRE] [A
VENDRE] [A VENDRE] [A VENDRE]]
```

Et modifions PERSPECTIVE en une scène de rue où le nom de l'habitant s'inscrit au moment où sa maison se dessine.

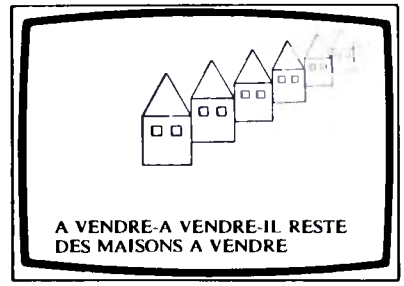
Tout d'abord il nous faut adjoindre à PERSP la liste des propriétaires :

```
POUR PERSP1 :X :LISTE
SI :X < 10 [STOP]                    Si : X inférieur à 10 STOP
FCFT 4 FCT 5                         Fixe la couleur du fond du texte à bleu et la cou-
TAPE PREM :LISTE TAPE [-]            leur du texte à violet. Affiche le premier membre
MAISON :X LC AVANCE 2 * ( :X / 5 ) FCA!   de la liste sans passer à
P 0 BC                                    la ligne.
PERSP1 :X - 10 SP :LISTE             Maison et position pour la suivante
FIN                                  Appel récursif en diminuant côté et ôtant le pre-
                                     mier membre de la liste.
```

Et la réalisation de la rue avec le nom des habitants se fera par :

```
POUR RUE
FEN
ME 2
LC FPOS [-100 -80] BC                MEMBRE? teste si [A VENDRE]
PERSP1 60 :PROPRIO                   appartient à la liste :PROPRIO
SI MEMBRE? [A VENDRE] :PROPRIO [TAPE [!
IL RESTE DES MAISONS A VENDRE]] [TAPE !
[TOUTES LES MAISONS SONT OCCUPEES]]
FIN

?RUE
```



Une rue qu'il serait agréable d'habiter... d'ailleurs, il reste des maisons à vendre !

A suivre...

Penser à ce qu'on a fait

1. Notion d'opération

Les primitives que nous venons d'introduire effectuent un travail "productif" sur les nombres, les mots ou les listes. Ce sont des primitives qui transforment une ou plusieurs données en une seule nouvelle donnée d'où leur nom d'**opération.** Le résultat ou valeur rendue doit alors être la donnée d'une nouvelle procédure.

Par exemple, prenons les données 2 et 3 sur lesquelles nous faisons agir la primitive SOMME le résultat est 5. On dit que SOMME 2 3 rend la somme de ses deux données.

Le résultat rendu par SOMME 2 3 peut alors être la donnée de ECRIS ou AVANCE ou la deuxième donnée de PROD 6 etc.

PREM [[ROSE MARGUERITE] FLEUR] rend [ROSE MARGUERITE] qui peut être à son tour la donnée de PREM ; PREM [ROSE MARGUERITE] rend ROSE qui à son tour peut devenir la donnée de PREM; PREM "ROSE rend R qui peut à son tour... mais PREM "R rend R...

Remarquons que SOMME et + rendent la même valeur, PROD et * également ainsi que DIFF et − ou DIV et /

Les deux données de SOMME, PROD, DIV, QUOT ou DIFF suivent cette primitive, ce sont des opérations **préfixées** alors que les symboles +, −, / et * sont entre les deux données : ce sont des opérations **infixées.**
Comme nous l'avons déjà vu, **l'ordinateur effectue les opérations infixées en respectant leur priorité.** La division est effectuée en premier, ensuite la multiplication, puis la soustraction et enfin l'addition.

2. Les expressions conditionnelles

Ce sont les phrases Logo libellées de la façon suivante :
SI (C) [*actions*1] [*actions*2]

ce qui signifie : lorsque la condition (C) est vraie on exécute la liste d'instruction (actions1), sinon on exécute la liste d'instructions (actions2) puis si (actions1) ou (actions2) ne terminent pas l'exécution, on passe aux instructions écrites à la ligne suivante. Rappelez-vous que toutes les données de SI doivent être écrites sur la *même* ligne Logo c'est-à-dire sans appuyer sur ENTREE.

Un cas particulier de cette phrase Logo est le test d'arrêt que nous avons vu au Chapitre 5.

3. Du rôle de PREM et SP

(pour PREMier et SaufPremier)

Reprenons la procédure DICTE :

Comment épeler le mot "BON

On commence par épeler la PREMière lettre de "BON
puis on DICTE le mot "ON qui est le SP de "BON

Pour DICTE du mot "ON on épelle la PREMière lettre de "ON puis on DICTE le mot "N qui est le SP de "ON.

Pour DICTE "N on épelle la première lettre de "N.
Puis on a fini puisqu'il n'y a plus de lettre : le mot vide est le SP de "N.

Les trois actions sont identiques, il n'y a que le mot sur lequel elles opèrent qui change. Cette **action** sur un mot quelconque consiste à épeler la PREMière lettre du mot et a recommencer avec le SP de ce mot jusqu'à ce que le mot se soit vidé de ses lettres. C'est exactement ce que décrit la procédure récursive DICTE.

Remarquons que les primitives PREM et SP admettent pour donnée soit un mot soit une liste et DICTE aussi.

Pour réaliser un dessin, nous avons vu qu'il existe deux stratégies :

a) Reproduire le dessin en pas à pas (mode direct).

b) Chercher les formes élémentaires qui composent le dessin ; leur associer des procédures et reconstruire le dessin.

La deuxième stratégie nécessite une certaine analyse du dessin.

Lorsque nous voulons décrire une opération sur les listes quelle est la stratégie à adopter ?

La primitive qui permet de transformer une liste en une liste plus courte est SP (ou SD c'est-à-dire SaufDernier).

Toute action sur une liste sera analysée de façon à faire apparaître une action identique sur la liste réduite (SP ou SD de cette liste).

L'action sera alors décrite comme un "automatisme" : expliciter ce qu'il faut faire du PREMier (DER) de la liste et recommencer avec SP (SD) de cette liste.
Cette analyse conduit à l'écriture d'une procédure récursive structurée de la façon suivante :

— action à effectuer si la liste est vide,

— ce qu'il faut faire du PREM (ou de DER) de la liste,

— rappel de la procédure avec pour donnée le SP (ou SD) de la liste.

Chapitre 8

Des procédures de dialogue

Dans ce chapitre nous présentons l'utilisation des primitives LL et LISCAR, deux primitives qui vous permettront d'écrire des procédures de dialogue.

Faire

Demandons à l'ordinateur de cesser toute activité pour être attentif à un message que nous allons lui donner :

```
?LL
```

Le ? a disparu sans faire place à un autre signe. Tapons notre message :

```
TOUT VA BIEN
QUE FAIRE DE [TOUT VA BIEN]
```

C'est la situation dans laquelle nous étions au chapitre précédent. LL est donc une primitive qui "rend" une phrase tapée au clavier.

```
?ECRIS LL
TOUT VA BIEN                    ce que nous tapons
TOUT VA BIEN                    ce que l'ordinateur affiche
```

Demandons de garder cette phrase sous le nom de "MESSAGE" :

```
?DONNE "MESSAGE LL
```

Tapons donc le message :

```
TOUT VA BIEN
```

Vérifions que l'ordre a été bien enregistré :

```
?ECRIS :MESSAGE
TOUT VA BIEN
```

C'est ce que nous voulions.
Ce que nous avons tapé au clavier est maintenant le contenu de MESSAGE.
Quelle est la structure de :MESSAGE

```
?ECRIS LISTE? :MESSAGE
VRAI
```

LISTE? rend VRAI si sa donnée est une liste, FAUX sinon.
Ici, :MESSAGE est une liste (comme l'indique LL : lis liste !) Nous

pouvons lui appliquer tous les traitements sur les listes que nous avons vu au chapitre précédent.

Peut-on savoir si une liste contient un mot (ou une liste) ?

```
?ECRIS MEMBRE? "BIEN :MESSAGE
VRAI

?ECRIS MEMBRE? "MAL :MESSAGE
FAUX
```

Une autre façon de lire non pas toute une phrase mais un seul caractère tapé au clavier :

```
?ECRIS LISCAR
(Appuyez sur A)
A                          ce que l'ordinateur affiche
```

inutile même de taper sur ENTREE !
Nous voici donc prêt pour écrire quelques dialogues.

Apprendre à la machine comment faire

1. Prendre contact

Bien souvent un ordinateur commence par demander le prénom de son interlocuteur, lui souhaite la bienvenue et éventuellement lui demande de faire son choix parmi un éventail de programmes.
Écrivons cela en Logo : utilisons la procédure RUE du Chapitre 5 à condition évidemment que RUE se trouve bien en mémoire centrale...

```
POUR BIENVENUE
ECRIS [COMMENT VOUS APPELEZ-VOUS?]
ECRIS PH "BONJOUR LL
ECRIS [VOULEZ-VOUS VOIR UNE BELLE RUE?!
 ( O/N )]
SI EGAL? LISCAR "O [RUE] [ECRIS [TANT !
PIS, A BIENTOT...]]
FIN
```

Essayons

```
?BIENVENUE
COMMENT VOUS APPELEZ-VOUS?
ALICE
BONJOUR ALICE VOULEZ-VOUS VOIR CETTE BELLE RUE? (O/N)
```

Et ALICE suit l'ordinateur au pays des merveilles!!!

2. Petite devinette

Connaissez-vous ce dialogue d'enfant :

— Combien font 3 et 2
— 5
— Hé non ! 32

ou bien :
— Combien font 3 et 2
— 32
— Hé non ! 5

Ce qui s'écrit en Logo :

```
POUR DIALOGUE
ECRIS [COMBIEN FONT 3 ET 2 ?]
SI EGAL? LL [5] [ECRIS [EH NON! 32!]] [!
ECRIS [EH NON! 5!]]
FIN

?DIALOGUE
```

Pas moyen de gagner !...
Ce dialogue peut s'améliorer en tirant les chiffres au hasard et en vérifiant bien que l'interlocuteur ne répond pas n'importe quoi !

```
POUR DIALOGUEBIS
DONNE "N1 1 + HASARD 9      pour éviter 0 en premier chiffre
DONNE "N2 1 + HASARD 9
EC PH PH PH [COMBIEN FONT] :N1 "ET :N2 !
DONNE "REP LL
SI EGAL? PREM :REP :N1 + :N2 [ECRIS PH !
[EH NON!] MOT :N1 :N2] [SI EGAL? PREM :!
REP MOT :N1 :N2 [EC PH [EH NON!] :N1 + !
:N2] [EC [VOUS REPONDEZ N'IMPORTE QUOI!!
ON RECOMMENCE..] DIALOGUEBIS]]
FIN


?DIALOGUEBIS
```

Toujours perdant !

3. Déplacer la tortue avec les touches

Pour un petit enfant on peut imaginer de mettre des gommettes de couleures différentes sur 4 touches qui feront déplacer la tortue.

Supposons que ces 4 touches sont A R D G
la touche A devra correspondre à AVANCE 10
la touche R devra correspondre à RECULE 10
la touche G devra correspondre à TG 30
la touche D devra correspondre à TD 30

Voici le programme qui permet ceci :

```
POUR DEPLACE
DONNE "C LISCAR
SI EGAL? :C "A [AVANCE 10]
SI EGAL? :C "R [RECULE 10]
SI EGAL? :C "G [TG 30]
SI EGAL? :C "D [TD 30]
DEPLACE
FIN

?MT DEPLACE
```

A vous de la guider !... Vous pouvez augmenter ce programme par des touches pour LC, BC éventuellement une touche de petit pas...

4. ...Et la maison !

Reprenons la procédure PERSP du chapitre 5 et toutes les procédures qu'elle utilise :

```
POUR PERSP :X
SI :X < 10 [STOP]
BC MAISON :X                          Action 1
LC AVANCE 2 * ( :X / 5 ) FCAP 0       Action 2
PERSP :X - 10                         Action 3
FIN
```

Modifions cette procédure de façon à attribuer des habitants à chaque maison au fur et à mesure de leur construction. Nous allons donc créer une liste des habitants.

Après chaque dessin de maison nous demanderons "qui habite cette

maison ?" et nous rangerons le nom des occupants à la suite de la liste des habitants.

Afin de conserver groupés tous les occupants d'une même maison, nous utiliserons la primitive MD pour agrandir la liste des habitants. Le k-ième élément de :HABITANTS sera alors représenté par les habitants de la k-ième maison.
Ceci se situe entre l'action 2 et l'action 3 de PERSP :

```
POUR PERSP1 :X
SI :X < 10 [STOP]
BC MAISON :X
LC AVANCE 2 * ( :X / 5 ) FCAP 0
ECRIS [QUI HABITE ICI?]
DONNE "HABITANTS MD LL :HABITANTS
PERSP1 :X - 10
FIN
```

Nous ne pouvons pas demander tout de suite PERSP1. En effet :
— la tortue n'est pas positionnée pour commencer le dessin.
— :HABITANTS n'a encore reçu aucune chose (n'a pas été initialisé).

Écrivons une procédure qui prend en compte ces remarques et appelle directement PERSP1 :

```
POUR HABITERUE
FEN
ME 2
LC FPOS [-100 -80] FCAP 0 FCFG 0
DONNE "HABITANTS []
PERSP1 60
FIN
```

Tout est prêt, nous pouvons personnaliser notre rue :

```
?HABITERUE
QUI HABITE ICI?
ANNE ET ANTOINE
QUI HABITE ICI?
JEAN ET JEANINE
QUI HABITE ICI?
SOPHIE
QUI HABITE ICI?
PIERRE
QUI HABITE ICI?
CLAUDE ET CLAUDINE
QUI HABITE ICI?
SYLVAIN
```

Nous voilà prêts pour écrire le bottin de cette rue.

La rue de la tortue n'a que des numéros pairs et 6 maisons ; ses habitants sont dans la liste :HABITANTS. Tout d'abord écrivons la procédure qui permet d'automatiser l'écriture du bottin :

```
POUR BOT :N :LISTE                    :N contiendra la place de la maison
SI :N > 6 [STOP]                      :LISTE contiendra la liste des habitants de la rue
EC PH PH 2 * :N [RUE DE LA TORTUE] PREM!
 :LISTE
BOT :N + 1 SP :LISTE
FIN
```

L'écriture du BOTTIN débute à la première maison, pour la liste :HABITANTS

```
POUR BOTTIN
BOT 1 :HABITANTS
FIN
```

```
?BOTTIN

 2 RUE DE LA TORTUE ANNE ET ANTOINE
 4 RUE DE LA TORTUE JEAN ET JEANINE
 6 RUE DE LA TORTUE SOPHIE
 8 RUE DE LA TORTUE PIERRE
10 RUE DE LA TORTUE CLAUDE ET CLAUDINE
12 RUE DE LA TORTUE SYLVAIN
```

A suivre...

Penser à ce qu'on a fait

1. La primitive LL

Cette primitive permet de faire intervenir dans le déroulement d'un programme un message tapé au clavier.

Dès qu'on appuie sur la touche ENTREE, l'ordinateur lit ce message et l'oublie juste après. Il est possible de demander un traitement du message pendant la lecture comme dans le cas :

```
ECRIS PH "BONJOUR LL
```

ensuite, ce qui a été lu est oublié.

Pour **stocker** un message il faut lui donner un nom.

```
DONNE "REPONSE LL
```

REPONSE contient alors la liste tapée au clavier.Même si ce qui est tapé est un seul mot, il est considéré comme une liste ayant un seul membre. En particulier si on demande par exemple le prénom d'une personne, ce prénom sera alors le PREM de LL ou de :REPONSE ce qui explique l'utilisation de PREM :REP dans le DIALOGUEBIS.

Donnons un autre exemple :

```
POUR INSCRIT
ECRIS [QUEL EST VOTRE PRENOM?]
DONNE "PRENOM LL
FIN
```

Après exécution de cette procédure, si vous avez donné comme prénom PIERRE, :PRENOM est alors [PIERRE]. En fait votre prénom est le premier membre de la liste [PIERRE] c'est-à-dire de :PRENOM.

2. La primitive LISCAR

Cette primitive LISCAR permet de lire immédiatement *un seul* caractère tapé au clavier.

```
DONNE "C LISCAR
```

Si nous tapons A alors :C est "A, le mot formé d'une seule lettre A. Cependant le caractère A n'est pas affiché sur l'écran.

LISCAR est souvent utilisé dans les procédures demandant une seule réponse (O/N) = oui/non) et dès que la réponse est donnée, la procédure continue son déroulement (inutile de taper sur ENTREE).

3. La primitive MD

C'est elle qui permet d'agrandir une liste en conservant la structure de liste ou de mot de ce qui est ajouté.

MD liste1 liste2 met la liste1 en dernier élément de la liste2. Par exemple :

```
MD [IL FAIT BEAU] [AUJOURD'HUI] est [AUJOURD'HUI [IL FAIT BEAU]]
```

Alors que

```
PH [IL FAIT BEAU] [AUJOURD'HUI] est [IL FAIT BEAU AUJOURD'HUI]
```

Remarque : nous avons déjà vu le principe de l'écriture conditionnelle au Chapitre 5.

Regardons le déroulement de cette procédure :

```
POUR DEPLACE
DONNE "C LISCAR                    Tapons R
SI EGAL? :C "A [AVANCE 10]         ce n'est pas le cas
SI EGAL? :C "R [RECULE 10]         c'est le cas, la tortue recule de 10 pas
SI EGAL? :C "G [TG 30]             ce n'est pas le cas
SI EGAL? :C "D [TD 30]             ce n'est pas le cas
DEPLACE                            on recommence
FIN
```

Et si nous ne tapons aucun membre de [A R G D] nous recommençons DEPLACE...

Appuyez sur [CNT] [C] pour sortir du déroulement de cette procédure. On peut aussi définir dans DEPLACE une touche d'arrêt. Par exemple :

```
SI EGAL? :C "F [STOP]
```

Chapitre 9

Les procédures-opérations

Dans ce chapitre, vous verrons comment construire des procédures-opérations qui rendent un résultat.

Faire

Maintenant nous connaissons des primitives qui rendent un résultat (Chapitres 7 et 8).

Considérons la procédure :

```
POUR AUCARRE :N
ECRIS PROD :N :N
FIN
```

cette procédure fait afficher le carré d'un nombre.

Si nous voulions utiliser le résultat du calcul effectué par AUCARRE pour une autre action (AVANCE par exemple), ceci ne serait pas possible. Le résultat de PROD :N :N a été immédiatement utilisé comme donnée de ECRIS à l'intérieur de la procédure.

En Logo, il y a une primitive qui permet de conserver le résultat de l'action d'une procédure.

C'est la primitive RENDS

Une procédure définie au moyen de RENDS est alors équivalente à une opération. Ce qui est produit par la procédure, pourra par la suite être utilisé comme donnée d'une autre primitive ou procédure.

Nous pouvons maintenant modifier la procédure AUCARRE de façon à ce qu'elle calcule le carré d'un nombre.

```
POUR AUCARRE :N
RENDS :N * :N
FIN
```

AUCARRE 2Ø par exemple, peut alors devenir aussi bien la donnée de ECRIS que de AVANCE ou de PREM etc., elle peut aussi être combinée avec d'autres primitives ou procédures-opérations.

Essayons :

```
?ECRIS AUCARRE 5
25

?ECRIS AUCARRE SOMME 3 4
49
?ECRIS AUCARRE AUCARRE 2
16

?ECRIS AUCARRE 4 + AUCARRE 5
841
```

Ce qui revient à l'écriture

```
?ECRIS AUCARRE (4 + AUCARRE 5)
841
```

Qui est différent de :

```
?ECRIS SOMME AUCARRE 4 AUCARRE 5
41

?AVANCE PREM AUCARRE 9
```

La tortue avance de 8 pas.

Apprendre à la machine comment faire

1. La moyenne

Nous pouvons définir MOYENNE comme étant une procédure qui "rend" la moyenne de deux nombres.

```
POUR MOYENNE :N1 :N2
RENDS ( :N1 + :N2 ) / 2
FIN
```

Essayons-la

```
?ECRIS MOYENNE 2Ø 1Ø
15

?ECRIS MOYENNE AUCARRE 4 AUCARRE 2
1Ø
```

Nous pouvons définir des procédures qui calculent la moyenne de deux carrés ou le carré d'une moyenne.

```
POUR MOYENNECARRE :N1 :N2
RENDS MOYENNE AUCARRE :N1 AUCARRE :N2
FIN

?ECRIS MOYENNECARRE 10 4
58

POUR CARREMOYENNE :N1 :N2
RENDS AUCARRE MOYENNE :N1 :N2
FIN

?ECRIS CARREMOYENNE 10 4
49
```

2. Calcul du nombre de lettres d'un mot

Reprenons le schéma donné au Chapitre 7 :
Si le mot est vide il a Ø lettre
Et il a 1 lettre de plus que ce mot tronqué de sa première lettre.

```
POUR NLETTRES :MOT
SI VIDE? :MOT [RENDS Ø] [RENDS SOMME 1 !
NLETTRES SP :MOT]
FIN

?ECRIS NLETTRES "NABUCODONOSOR
13
```

3. Remplacer un mot d'une phrase par un autre

Pour remplacer :MOT1 d'une liste par :MOT2 il faut à chaque fois que nous rencontrons :MOT1 l'enlever et mettre :MOT2 à sa place. Ce qui peut être décrit par la procédure :

```
POUR REMPLACE :LISTE :MOT1 :MOT2
SI VIDE? :LISTE [RENDS []]
SI EGAL? PREM :LISTE :MOT1 [RENDS PH :M!
OT2 REMPLACE SP :LISTE :MOT1 :MOT2] [RE!
NDS PH PREM :LISTE REMPLACE SP :LISTE :!
MOT1 :MOT2]
FIN
```

Essayons :

```
?ECRIS REMPLACE [LE CIEL EST GRIS] "GRIS "BLEU
LE CIEL EST BLEU
```

4. Retenir les déplacements de la tortue

Reprenons la procédure DEPLACE du Chapitre 8

```
POUR DEPLACE
DONNE "C LISCAR
SI EGAL? :C "A [AVANCE 10]
SI EGAL? :C "R [RECULE 10]
SI EGAL? :C "G [TG 30]
SI EGAL? :C "D [TD 30]
DEPLACE
FIN
```

modifions-la afin que :

— DEPLACE produise la liste des déplacements
— Ne soit pas prise en compte toute lettre frappée autre que A, R, G, ou D
— Dès que F est tapée la procédure s'arrête.

DEPLACE va devenir la procédure qui **rend** la liste des déplacements sans tenir compte des déplacements "impossibles" (autres que A,R,G,D) et qui stoppe dès que F est tapé.

Ce qui peut se programmer de la façon suivante :

```
POUR DEPLACE
DONNE "C LISCAR
SI NON MEMBRE? :C [A R D G F] [RENDS DE!
PLACE]
SI EGAL? :C "A [AV 10]
SI EGAL? :C "R [RECULE 10]
SI EGAL? :C "D [TD 30]
SI EGAL? :C "G [TG 30]
SI EGAL? :C "F [RENDS "]
RENDS PH :C DEPLACE
FIN
```

DEPLACE produit alors la liste des déplacements demandés.

Essayons

```
?ECRIS DEPLACE
A   ce que nous tapons
B   ce que nous tapons
R   ce que nous tapons
G   ce que nous tapons
F   ce que nous tapons
A R G F ce que l'ordinateur affiche
```

5. Travail sur les nombres

Nous allons faire le produit de tous les nombres entiers plus petits qu'un nombre donné. C'est ce qu'on appelle en mathématiques une factorielle. La factorielle s'écrit **n!** et **n** peut être n'importe quel nombre entier positif. Par exemple, 4! = 1*2*3*4 = 24. Par définition, Ø! = 1

Voici comment écrire ceci en Logo :

```
POUR FACT :N
SI PLP? :N Ø [RENDS [PAS DE NOMBRE NEG!
ATIF]]
SI EGAL? :N Ø [RENDS 1] [RENDS PROD :N!
 FACT :N - 1]
FIN
```

PLP signifie plus petit

Par exemple :

```
?ECRIS FACT 5
12Ø
```

6. ...Et la maison !

Comment trouver l'adresse exacte d'une personne habitant rue de la Tortue ?

Il suffit de connaître la position qu'occupe cette personne parmi la liste des habitants pour avoir son adresse exacte.

Comment connaître la position d'un membre (mot ou liste) dans une liste ? Prenons par exemple un **mot.**

— Si ce mot est le premier de la liste il occupe la position n° 1 ! sinon il occupe la position dont le numéro se calcule par 1 + sa position dans le SP de la liste :

```
POUR POSITION :MOT :LISTE
SI EGAL? PREM :LISTE :MOT [RENDS 1] [RE!
NDS 1 + POSITION :MOT SP :LISTE]
FIN
```

Par exemple :

```
?ECRIS POSITION "SOPHIE [JEAN ANNE SOPHIE PIERRE]
3
```

— Si ce mot n'est pas dans la liste inutile de le chercher ! Ceci sera à tester avant de demander la POSITION.

Et pour demander de commencer une recherche :

```
POUR RECHERCHE
ECRIS [QUI CHERCHEZ-VOUS?]
DONNE "REP LL
SI NON MEMBRE? :REP :HABITANTS [ECRIS [!
PAS DANS CETTE RUE]] [ECRIS PH 2 * POSI!
TION :REP :HABITANTS [RUE DE LA TORTUE]!
]
FIN
```

```
?RECHERCHE
```

A vous de compléter cette procédure que pour si vous ne connaissez qu'ANNE vous puissiez aboutir à la conclusion :

```
ANNE ET ANTOINE
2 RUE DE LA TORTUE
```

A suivre...

Penser à ce qu'on a fait

Les **opérations** sont des procédures qui "calculent" et le résultat de leur action est mis en évidence par la primitive RENDS. Ces procédures peuvent alors devenir des données pour d'autres procédures qui aussi à leur tour... C'est le début de l'engrenage !

1. La primitive RENDS

C'est la primitive qui dit à l'ordinateur de garder en mémoire une donnée ou le résultat d'une action. Si ce résultat n'est pas immédiatement utilisé comme donnée pour des actions, l'ordinateur affichera QUE FAIRE DE (Résultat). RENDS s'utilise uniquement dans une procédure.
Les instructions qui suivent RENDS sont obligatoirement elles-mêmes des procédures-opérations ou des primitives-opérations.

```
RENDS ECRIS 10
```

lors de l'exécution de cette procédure, l'ordinateur affichera 10 et le message :

```
PAS ASSEZ DE DONNEES POUR RENDS
```

2. Les parenthèses

Les parenthèses servent à indiquer une organisation des opérations différente de celle qui est faite "naturellement" par la machine.

On les utilise aussi pour rendre plus claire et plus lisible une écriture compliquée, sans vouloir changer l'ordre d'exécution des actions (pour distinguer les données de PH par exemple).

3. La récursivité dans les procédures-opérations

1. Reprenons pas à pas la procédure NLETTRE appliquée au mot "BON

NLETTRE "BON = 1 + NLETTRE "ON
NLETTRE "ON = 1 + NLETTRE "N
NLETTRE "N = 1 + NLETTRE "
NLETTRE " = 0 [STOP] et là l'ordinateur a toutes les informations pour calculer
NLETTRE "N = 1 + 0 = 1
NLETTRE "ON = 1 + 1 = 2
NLETTRE "BON = 1 + 2 = 3
NLETTRE "BON est 3

Notez l'utilisation de RENDS dans une procédure récursive :
A chaque appel récursif, ce qui est produit par la procédure est prêt à être utilisé. Cette utilisation s'effectue au moment du dernier appel.
Le processus est le même pour le calcul de la factorielle d'un nombre : procédure FACT.

2. Reprenons la procédure REMPLACE de ce chapitre et analysons-la suivant le schéma donné au Chapitre 7 :

```
POUR REMPLACE :LISTE :MOT1 :MOT2
SI VIDE? :LISTE [RENDS []]
SI EGAL? PREM :LISTE :MOT1 [RENDS PH :M!
OT2 REMPLACE SP :LISTE :MOT1 :MOT2] [RE!
NDS PH PREM :LISTE REMPLACE SP :LISTE :!
MOT1 :MOT2]
FIN
```

— Si la liste est **vide** il n'y a rien à faire c'est-à-dire qu'on conserve cette liste vide et le travail est terminé :

```
SI VIDE? :LISTE [RENDS []]
```

— Si le premier de la liste est :MOT1 il doit être remplacé par :MOT2 sinon il est conservé et le reste de la liste (SP) subit à son tour le même traitement.

Si le premier mot de la liste est :MOT1 il faut mettre :MOT2 à sa place et le reste de la liste (SP) subit le même traitement :

```
[RENDS PH :MOT2 REMPLACE SP :LISTE]
```

4. Les prédicats et les opérateurs logiques

On appelle ainsi les primitives qui retournent VRAI ou FAUX. Nous avons déjà rencontrées :

```
VIDE?
MEMBRE?
EGAL?
```

Vous pouvez même définir vos propres opérateurs logiques. Ces opérateurs sont alors utilisés dans des tests conditionnels (voir exemple 6). Ce test conditionnel peut s'écrire :

SI (Op logique) [*actions1*] [*actions2*]

Au lieu d'écrire :

SI EGAL? (Op logique) VRAI [*actions1*] [*actions2*]

Pour éviter l'emploi trop lourd des crochets il est parfois plus simple de donner en premier la liste des actions à accomplir lorsque le résultat du test est FAUX ce qui s'écrit :

SI NON (Op logique) [*actions2*] qui est équivalent à :
SI EGAL? (Op logique) FAUX [*actions2*]

Signalons également les opérateurs ET et OU :
Ce sont des opérateurs logiques préfixés opérant sur 2 tests :

ET *condition1 condition2* rendent VRAI si *condition1 et condition2* rendent VRAI ensemble.
OU *condition1 condition2* rend FAUX seulement si *condition1 et condition2* rendent FAUX.

Nous utiliserons ces opérateurs au cours de l'élaboration des projets du Chapitre 12.

Chapitre 10

Le crayon optique et les manettes de jeux

1. Le crayon optique

Si votre ordinateur est muni d'un crayon optique, celui-ci vous permet de désigner une position sur l'écran.

Le crayon optique est prêt à fonctionner, dès que vous chargez le langage Logo dans la mémoire de votre ordinateur. Pour l'utiliser, il est bon de procéder à son réglage. Si vous possédez un TO7, ou un TO7-70, réglez le crayon optique avant la mise en œuvre de LOGO.
Si vous possédez un MO5, utilisez la primitive REGLE et désignez alors la barre verticale au centre de l'écran avec le crayon optique.

Une primitive permet d'obtenir les coordonnées du point de l'écran sur lequel "pointe" le crayon. (Pour pointer, il faut appuyer la pointe du crayon sur l'écran).

Cette primitive est POSOPT ; ce qui veut dire POSition du crayon OPTique. Ainsi,

```
?ECRIS POSOPT
```

affiche sur l'écran la liste des coordonnées de l'endroit pointé

```
?FPOS POSOPT
```

place la tortue aux coordonnées pointées par le crayon optique

Une deuxième primitive se rapportant au crayon optique est CONTACT?. Celle-ci rend VRAI si la pointe du crayon optique est enfoncée, FAUX sinon.

1. Nous pouvons utiliser l'écran comme une feuille de papier sur laquelle nous écrirons.

A chaque instant nous allons positionner la tortue sur le point désigné par le crayon optique.

A chaque fois que le crayon optique ne sera plus en **contact**, nous aurons la possibilité de changer de couleur.

Pour cela, dès que le crayon optique n'est plus en contact avec l'écran, faisons changer la couleur du crayon matérialisée par la couleur de la tortue.

Lorsque la position du crayon optique n'est pas un point du champ, le résultat de POSOPT est la liste vide. Il faut donc à chaque moment analyser le résultat de POSOPT pour ne pas être dans ce cas.

Nous allons donc créer une procédure DESSINE qui ne déplace la tortue que si le crayon est dans le champ. Et la procédure DESSIN permet de dessiner quand le crayon optique pointe sur l'écran ; sinon la couleur du crayon de la tortue change.

```
POUR DESSINE :LISTE
SI VIDE? :LISTE [STOP] [FPOS :LISTE]
FIN


POUR DESSIN
SI CONTACT? [DESSINE POSOPT BC] [LC FCC!
 CC + 1]
DESSIN
FIN
```

2. Saisie d'une zone de l'écran

Écrivons une procédure qui écrive le nom de dessins qui se trouvent sur l'écran.
Supposons qu'il y ait 4 dessins disposés de la façon suivante :

un carré	un cercle
un hexagone	un triangle

Il nous faut donc une procédure qui réalise le dessin et appelle une procédure de SAISIE.

Note :
Plus l'écran sera lumineux et plus il sera facile de relever ses coordonnées.
Sur un fond très sombre, vous aurez certainement des problèmes (évitez le noir et le rouge).

```
POUR PLANCHE
ME 2
CT
LC FPOS [-80 50]
BC REPETE 4 [AV 40 TD 90]
LC FPOS [80 50]
BC REPETE 180 [AV 1 TD 2]
LC FPOS [-80 -50]
BC REPETE 6 [AV 20 TD 60]
LC FPOS [80 -50]
BC REPETE 3 [AV 30 TD 120]
SAISIE
FIN
```

La procédure de SAISIE analyse le contenu de POSOPT lorsqu'il y a contact :

```
POUR SAISIE
SI CONTACT? [ANALYSE POSOPT]
SAISIE
FIN
```

Enfin l'analyse de POSOPT :

```
POUR ANALYSE :LISTE
SI VIDE? :LISTE [ECRIS [JE NE VOIS RIEN!
] STOP]
SI ET PREM :LISTE < 0 DER :LISTE > 0 [E!
CRIS [CECI EST UN CARRE] STOP]
SI ET PREM :LISTE < 0 DER :LISTE < 0 [E!
CRIS [CECI EST UN HEXAGONE] STOP]
SI ET PREM :LISTE > 0 DER :LISTE > 0 [E!
CRIS [CECI EST UN CERCLE]] [ECRIS [CECI!
 EST UN TRIANGLE]]
FIN
```

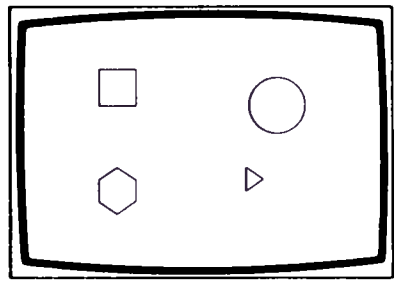
L'analyse décrit la situation suivante lorsque la liste n'est pas vide

```
PREM :LISTE > 0
DER :LISTE < 0

PREM :LISTE > 0
DER :LISTE > 0

PREM :LISTE < 0
DER : LISTE < 0
```

et si un des trois cas n'est pas vérifié, c'est que l'on se trouve dans le dernier quart de l'écran (en bas à droite).



2. Les manettes de jeux

Si vous possédez des manettes de jeux vous remarquerez que chaque manette est composée :

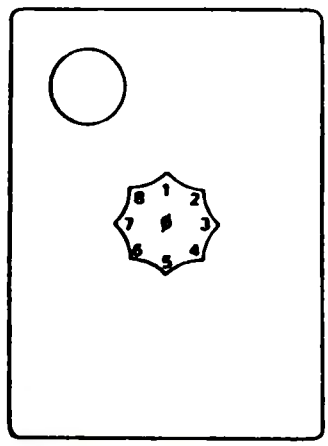
— d'un levier pouvant adopter 8 positions actives
— d'un bouton pouvant être pressé

Les primitives rendant compte de l'état du levier et du bouton sont MANETTE 1 (ou Ø) BOUTON? 1 (ou Ø).
Au repos :

```
?ECRIS MANETTE 1
Ø
?ECRIS BOUTON? 1
FAUX
```

Les positions du levier sont repérées de la façon suivante :

position	neutre	Ø
position	9Ø degrés	1
position	45 degrés	2
position	Ø degrés	3
position	— 45 degrés	4
position	— 9Ø degrés	5
position	— 135 degrés	6
position	18Ø degrés	7
position	135 degrés	8



et si le bouton est pressé

```
?ECRIS BOUTON? 1
VRAI
```

Nous pouvons par exemple utiliser ces manettes pour guider la tortue de la façon suivante

Manette 1
Son levier définit l'orientation de la tortue
Son bouton lève le crayon

Manette Ø
Son levier fixe la vitesse de la tortue
Son bouton baisse le crayon en changeant de couleur

```
POUR CARNAVAL
SI BOUTON? 1 [LC]
SI BOUTON? Ø [BC FCC CC + 1]
FCAP ((MANETTE 1) - 1 ) * 45
AVANCE (MANETTE Ø) * 5
CARNAVAL
FIN
```

et pour ne pas être limité par l'écran

```
?ENR
```

A vos manettes :

```
?CARNAVAL
```

Chapitre 11

La musique

Votre ordinateur possède un générateur de sons permettant la composition musicale sur une plage de cinq octaves. Le son étant restitué par le haut parleur de votre téléviseur vous pouvez en régler l'intensité en agissant sur le réglage "volume" de ce téléviseur.

En Logo la primitive qui vous permet de "jouer de la musique" est JOUE suivie d'une chaîne de caractères. Cette chaîne est votre composition musicale, c'est-à-dire que ses éléments sont :

— Les notes et les figures de notes éventuellement suivies d'une altération.
Par exemple :

```
JOUE "DOREMI
```

— Des indications sur le TIMBRE et sur l'OCTAVE sur lequel vous voulez jouer ainsi que le TEMPO c'est-à-dire la vitesse d'exécution du morceau et la DUREE, peuvent être précisés avant JOUE.
Si aucune indication n'est donnée, une valeur est prise par défaut.

Les notes

Ce sont les notes de la gamme appelées par leur nom : DO RE MI FA SO (pour SOL) LA SI PA (pour une pause) écrites comme dans l'exemple ci-dessus.

1. Les figures de notes

Elles sont données par la primitive DUREE (longueur) suivie d'un nombre suivant le code :

96 est une ronde
48 est une blanche
24 est une noire
12 est une croche
 6 est une double croche
 3 est une triple croche

Lorsque une figure de note est fixée, elle est valable pour toutes les notes qui la suivent.

Par exemple dans DUREE 96 JOUE "DORE DUREE 3 JOUE "MIFA
DO RE sont des rondes
MI FA sont des triples croches

Toutes les valeurs intermédiaires entre 3 et 96 sont admises.

DUREE 36 correspond à une noire pointée
DUREE 72 correspond à une blanche pointée
etc.

Si aucune figure de note n'est précisée c'est la dernière figure de note jouée qui est la valeur implicite et par défaut, c'est la noire.

2. L'octave

Elle est fixée par la primitive OCTAVE suivie d'un nombre compris entre 1 et 5 et concerne toutes les notes qui le suivent.

OCTAVE 1 est l'octave la plus grave
OCTAVE 5 est l'octave la plus aiguë

Écrivons les arpèges de la gamme DO majeur

```
POUR ARPEGE :X
SI :X < 1 [STOP]
OCTAVE :X JOUE "DOMISO
ARPEGE :X - 1
FIN
```

Essayons :

```
?ARPEGE 5
```

Il manque la descente.
Si aucune indication concernant l'octave n'est donnée l'ordinateur joue sur la dernière octave utilisée ou à défaut sur l'octave 4.

3. Les altérations

— Bémol <b abaisse le son de la note précédente d'un demi-ton
— dièse < # élève le son de la note précédente d'un demi-ton

```
POUR CHROMATIQUE
JOUE "DODO<#RERE<#MIFAFA<#SOSO<#LALA<#SIDO
FIN
```

Remarquez que le dernier DO est en fait le même que le premier. Aucune indication n'a été donnée pour passer à l'octave supérieur.

4. Le tempo

Il est donné par la primitive TEMPO à laquelle on doit attribuer un nombre compris entre 1 (rapide) et 255 (lent), du prestissimo au largo.

5. Le timbre

La primitive TIMBRE permet de modifier le son. TIMBRE Ø correspond à un son contenu et le son devient de plus en plus amorti pour des grandes valeurs.

Apprenons à la machine "j'ai du bon tabac" :

```
POUR TABAC
AIR
REFRAIN
REFRAIN
AIR
FIN
```

La procédure TABAC donne l'enchaînement de l'air et du refrain :

```
POUR REFRAIN
DUREE 24 JOUE "SO
DUREE 12 JOUE "SOFA
DUREE 24 JOUE "MI
DUREE 12 JOUE "REMI
DUREE 24 JOUE "FASO
DUREE 48 JOUE "RE
FIN
```

L'air lui-même est composé de 3 airs :

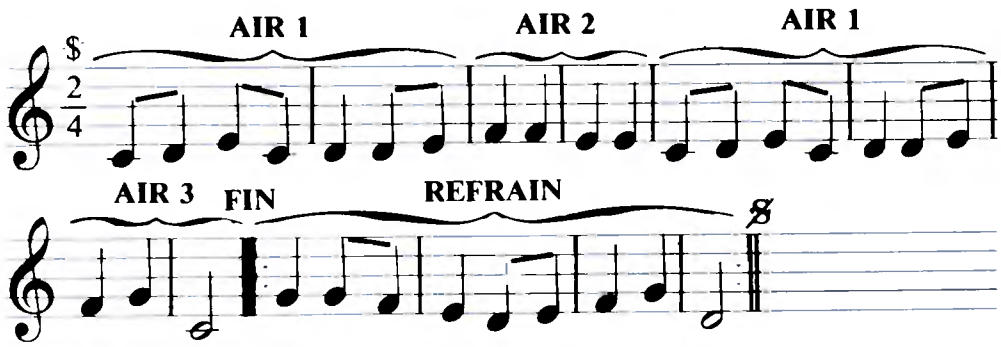
```
POUR AIR
AIR1
AIR2
AIR1
AIR3
FIN
```

Il ne reste plus qu'à écrire les procédures permettant de jouer AIR1, AIR2 et AIR3

```
POUR AIR1
DUREE 12 JOUE "DOREMIDO
DUREE 24 JOUE "RE
DUREE 12 JOUE "REMI
FIN
```

```
POUR AIR2
DUREE 24 JOUE "FAFAMIMI
FIN
```

```
POUR AIR3
DUREE 24 JOUE "FASO
DUREE 48 JOUE "DO
FIN
```



Nous pouvons également créer une procédure permettant de modifier le tempo, l'octave et le timbre :

```
POUR JOUER :OCT :TIMBRE :TEMPO
OCTAVE :OCT
TIMBRE :TIMBRE
TEMPO :TEMPO
TABAC
FIN
```

Par exemple :

?JOUER 2 Ø 8 est à peu près ce dont nous avons l'habitude.

Nous pouvons prendre des libertés par rapport au thème :

```
POUR TABAC1
TEMPO 8
OCTAVE 5 AIR
OCTAVE 4 REFRAIN REFRAIN
OCTAVE 5 AIR
FIN
```

ou encore intercaller dans les différents AIR des notes qui permettent de faire une variation sur le thème "j'ai du bon tabac"... ou encore intercaller avant chaque air le texte associé...

Chapitre 12

Quatre projets

Projet 1 : le portrait chinois

C'est le jeu dont le but est de découvrir un personnage célèbre à partir d'une description du type :
Si c'était ... ce serait ...

Donnons un exemple :
Si c'était une fleur ce serait un cactus
Si c'était un animal ce serait un cheval
Si c'était une couleur ce serait le jaune
Si c'était un pays ce serait l'Amérique

Qui est-ce ?
Avez-vous deviné ?... Il s'agit de Lucky Luke !

Nous allons essayer de "programmer" ce jeu afin :
— soit de jouer : l'ordinateur affiche les comparaisons et nous devons deviner de qui il s'agit ;
— soit d'élargir le jeu, c'est-à-dire de donner à l'ordinateur un nouveau personnage et les comparaisons qu'il nous évoque.

1. Pour jouer

Prenons par exemple comme éléments de comparaison :
une fleur, un animal, une couleur, un pays :

Chaque personnage sera donc défini par sa comparaison à une fleur, un animal, une couleur et un pays.

Nous allons créer des listes pour chaque élément de référence ainsi qu'une liste de personnages :

a) Création de listes

```
?DONNE "FLEUR [[UN CACTUS] [UNE ROSE]]
?DONNE "ANIMAL [[UN CHEVAL] [UNE LICORNE]]
?DONNE "COULEUR [[LE JAUNE] [LE POURPRE]]
?DONNE "PAYS [[L'AMERIQUE] [LA FRANCE]]
?DONNE "PERSONNAGE [[LUCKY LUKE] [RONSARD]]
```

Le premier élément de :PERSONNAGE a pour référence les premiers éléments des autres listes — le deuxième élément de :PERSONNAGE a pour référence le deuxième élément des autres listes et ainsi de suite (lorsque les listes s'agrandiront...)
Comme les mots **une fleur, un animal...** reviendront toujours dans les questions mettons-les dans une liste, ce qui nous permettra "d'automatiser" le processus.

```
?DONNE "REFERE [[UNE FLEUR] [UN ANIMAL] [UNE COULEUR] [UN PAYS]]
```

b) Écriture des procédures permettant de jouer

Le JEU dépend de la liste des références et du personnage choisi.
Le personnage étant un élément (tiré au hasard) de la liste :PERSONNAGE
Écrivons la procédure JEU :

```
POUR JEU :LISTE :I
SI VIDE? :LISTE [QUESTION :I 3 STOP]
EC PH PH [SI C'ETAIT] PREM :LISTE [CE S!
ERAIT] EC PH [.....]!
 ITEM :I CHOSE DER PREM :LISTE
EC []                                    Pour passer une ligne
JEU SP :LISTE :I
FIN
```

:LISTE contiendra :REFERE
:I contiendra un nombre tiré au hasard compris entre 1 et COMPTE :PERSONNAGE.

QUESTION est une procédure qui permettra de donner 3 réponses avant que l'ordinateur donne la réponse correcte.

Commentons plus en détail la troisième ligne en supposant que :LISTE est :REFERE et que :I contient 1.

PREM :REFERE est [UNE FLEUR]

donc PH [SI C'ETAIT] PREM :LISTE donne SI C'ETAIT UNE FLEUR et PH [SI C'ETAIT UNE FLEUR] [CE SERAIT] donne SI C'ETAIT UNE FLEUR CE SERAIT

DER PREM :REFERE est "FLEUR

La primitive CHOSE permet alors de considérer le mot "FLEUR comme un nom et CHOSE "FLEUR est :FLEUR. ITEM 1 CHOSE DER PREM :LISTE est donc ITEM 1 :FLEUR qui est ITEM 1 [[UN CACTUS]...]

Et l'exécution de la troisième ligne affiche :
SI C'ETAIT UNE FLEUR CE SERAIT UN CACTUS

Écrivons la procédure qui "lance" le JEU. Elle donne à I un nombre compris entre 1 et le nombre d'éléments de la liste PERSONNAGE, (c'est-à-dire COMPTE :PERSONNAGE) et à :LISTE la liste :REFERE

```
POUR JOUER
VT JEU :REFERE 1 + HASARD COMPTE :PERSO!
NNAGE
FIN
```

VT signifie vide le texte

QUESTION est une procédure qui accorde au joueur 3 réponses avant de lui donner la bonne. Evidemment QUESTION dépend toujours du même nombre tiré au hasard dans JOUER :

```
POUR QUESTION :I :N
SI EGAL? :N Ø [EC PH [C'ETAIT ...] ITEM!
 :I :PERSONNAGE REJOUE STOP]
ECRIS [QUI EST-CE ?]
SI EGAL? LL ITEM :I :PERSONNAGE [ECRIS !
[BIEN VU ! BRAVO !!] REJOUE] [ECRIS [EH!
 NON!] QUESTION :I :N - 1]
FIN
```

:I recevra la même valeur que le :I de JEU
:N est le nombre de fois où on peut donner une réponse.
Au départ :N contiendra 3 fois (voir JEU)
REJOUE une procédure permettant de relancer le jeu.

Enfin la procédure REJOUE relancera le jeu si le joueur le désire (réponse O) sinon le jeu s'arrêtera :

```
POUR REJOUE
ECRIS [ON RECOMMENCE? ( O/N )]
SI EGAL? LISCAR "O [DEPART] [ECRIS [AU !
REVOIR] STOP]
FIN
```

Essayons cette partie du jeu :

```
?JOUER
SI C'ETAIT UNE FLEUR CE SERAIT UNE ROSE
SI C'ETAIT UN ANIMAL CE SERAIT UNE LICORNE
SI C'ETAIT UNE COULEUR CE SERAIT LE POURPRE
SI C'ETAIT UN PAYS CE SERAIT LA FRANCE

QUI EST-CE ?
```

Si vous tapez RONSARD l'ordinateur affichera BIEN VU ! BRAVO !! puis : ON RECOMMENCE ? (O/N)

Si vous tapez O le jeu recommencera sinon l'ordinateur affichera : AU REVOIR et vous sortirez du jeu. Sinon vous avez droit à deux autres réponses avant qu'il ne vous donne celle qu'il connaît.

```
ET NON ! C'ETAIT RONSARD
ON RECOMMENCE ? (O/N)
```

2. Pour élargir le jeu

Il s'agit d'agrandir les listes :PERSONNAGE :FLEUR :ANIMAL :COULEUR :PAYS. En posant les questions :

SI C'ETAIT ... CE SERAIT ... et en rangeant la réponse dans la liste adéquate. Ce qui se fera d'une façon analogue à la procédure JEU compte tenu que la primitive MD permet d'insérer une nouvelle liste dans une liste précédemment constituée :

```
POUR ENCORE :LISTE     :LISTE recevra la "chose" nommée REFERE
SI VIDE? :LISTE [TAPE [DE QUEL PERSONNA!
GE S'AGIT-IL ?] DONNE "PERSONNAGE MD LL!
 :PERSONNAGE STOP]
TAPE PH PH [SI C'ETAIT] PREM :LISTE [CE!
 SERAIT ...]
DONNE DER PREM :LISTE MD LL CHOSE DER P!
REM :LISTE
ENCORE SP :LISTE
FIN
```

Essayons :

?ENCORE :REFERE et tapons :
SI C'ETAIT UNE FLEUR CE SERAIT ... UNE PATIENCE
SI C'ETAIT UN ANIMAL CE SERAIT ... UN RENARD
SI C'ETAIT UNE COULEUR CE SERAIT ... LE ROUGE
SI C'ETAIT UN PAYS CE SERAIT ... LA BELGIQUE
DE QUEL PERSONNAGE S'AGIT-IL? MAIGRET

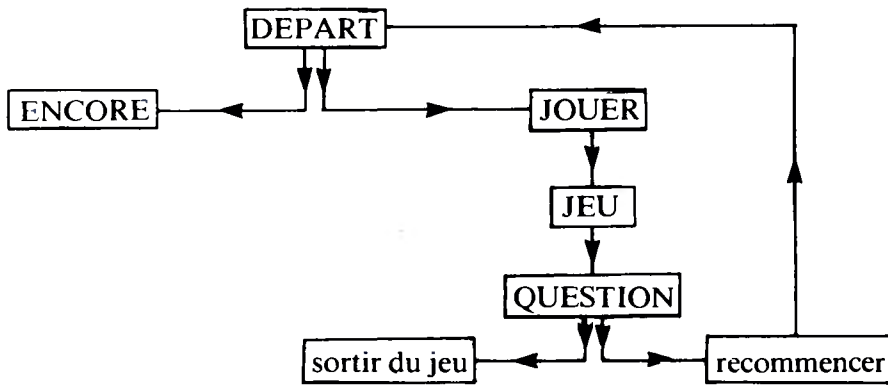
Vérifions :

?ECRIS :PERSONNAGE
[LUCKY LUKE] [RONSARD] [MAIGRET]

Et vous pouvez vérifier que les autres listes se sont également agrandies.

3. Les enchaînements

Il faut une procédure de DEPART qui permette soit de commencer le jeu soit de l'élargir et modifier les procédures ENCORE et QUESTION de façon à pouvoir revenir au DEPART. La "tour" des procédures se présentera de la façon suivante :



Écrivons DEPART :

```
POUR DEPART
VT
EC [VOULEZ-VOUS JOUER OU ELARGIR LE JEU!
 ?]
SI MEMBRE? "JOUER LL [JOUER] [EC [AVEZ-!
VOUS UN PERSONNAGE A DEFINIR ? ( O/N )]!
 SI EGAL? LISCAR "O [VT ENCORE :REFERE]!
 [DEPART]]
FIN
```

Cette procédure teste d'abord si le mot JOUER fait partie de la réponse à la question : VOULEZ-VOUS JOUER OU ELARGIR LE JEU ? c'est-à-dire qu'une réponse du type :
— Je veux jouer
— je ne veux pas élargir le jeu mais jouer
— jouer
appellera la procédure JOUER

Toute réponse ne comportant pas le mot jouer fera afficher
AVEZ-VOUS UN PERSONNAGE A DEFINIR ? (O/N)
et la procédure ENCORE ne sera appelé que si la réponse est Oui, sinon on revient au départ. Ce type de question permet d'éviter à l'utilisateur de s'engager dans une voie qu'il ne souhaitait pas vraiment mais qu'il a emprunté parce que la première question n'évoquait rien de très clair dans son esprit.

Et maintenant... à vous de jouer !

?DEPART

Projet 2 : promenons la tortue

Reprenons la procédure DEPLACE du Chapitre 9.

```
POUR DEPLACE
DONNE "C LISCAR
SI NON MEMBRE? :C [A R D G F] [RENDS DE!
PLACE]
SI EGAL? :C "A [AVANCE 1Ø]
SI EGAL? :C "R [RECULE 1Ø]
SI EGAL? :C "D [TD 3Ø]
SI EGAL? :C "G [TG 3Ø]
SI EGAL? :C "F [RENDS "]
RENDS PH :C DEPLACE
FIN
```

Cette procédure permet de créer une liste de déplacements du type [A D R G D D] où :

A correspond à AVANCE 1Ø
R correspond à RECULE 1Ø
D correspond à TD 3Ø
G correspond à TG 3Ø

Analysons cette liste de façon :

a) à éliminer G et D s'ils sont l'un à côté de l'autre. En effet tourner à gauche de 30 degrès, puis à droite de 30 degrès ou vice versa sont deux actions qui s'annulent.

b) à transformer le résultat des déplacements en une liste d'actions :

```
[AVANCE 10 TD 30 RECULE 10 TD 30]
```

Cette liste d'actions étant crée, pour refaire le dessin il suffira d'ordonner EXEC c'est-à-dire EXECute la liste d'actions.

1. Eliminer les lettres G et D d'une liste si elles sont écrites côte à côte

La stratégie que nous adoptons consiste à transporter membre par membre une liste dans un mot. Si le dernier transporté est D (ou G) et le suivant G (ou D) il faut les effacer.

Fixons une liste de référence qui nous permettra de tester les procédures ainsi définies :

```
?DONNE "REF [A D R G D D A D G]
```

et écrivons la procédure qui transformera cette liste en le mot :

ADRDA

```
POUR TRANS :LISTE :MOT
SI VIDE? :LISTE [RENDS :MOT]
SI VIDE? :MOT [RENDS TRANS SP :LISTE PR!
EM :LISTE]
SI ET MEMBRE? DER :MOT [D G] MEMBRE? PR!
EM :LISTE [D G] [SI EGAL? DER :MOT PREM!
 :LISTE [RENDS TRANS SP :LISTE MOT :MOT!
 PREM :LISTE] [RENDS TRANS SP :LISTE SD!
 :MOT]] [RENDS TRANS SP :LISTE MOT :MOT!
 PREM :LISTE]
FIN
```

Au début :LISTE contiendra SP :REF et :MOT contiendra "A c'est-à-dire PREM :REF
Supposons que :MOT est "D et le premier de :LISTE est "G. Après transformation :MOT sera vide, ce qui oblige à considérer le cas SI VIDE? :MOT [RENDS TRANS...

Et pour réduire une liste donnée :

```
POUR REDUIS :LISTE
RENDS TRANS SP :LISTE PREM :LISTE
FIN
```

```
?ECRIS REDUIS :REF
ADRDA
```

ce que nous voulions.

2. Transformer un mot ne contenant que les lettres A D G R en une suite d'actions

Il suffit d'écrire une procédure analogue à DEPLACE sachant qu'au lieu d'être entrée au clavier, la lettre sera lue dans le mot.

```
POUR INTERPRETE :MOT
SI VIDE? :MOT [RENDS []]
SI EGAL? PREM :MOT "A [RENDS PH [AVANCE!
 10] INTERPRETE SP :MOT]
SI EGAL? PREM :MOT "R [RENDS PH [RECULE!
 10] INTERPRETE SP :MOT]
SI EGAL? PREM :MOT "G [RENDS PH [TG 30]!
 INTERPRETE SP :MOT]
SI EGAL? PREM :MOT "D [RENDS PH [TD 30]!
 INTERPRETE SP :MOT]
FIN
```

Nous sommes maintenant prêts à guider la tortue à l'aide du clavier et à garder la liste des actions sous le nom DEPLACEMENT, à condition que DEPLACE soit en mémoire centrale (voir deuxième partie, Chapitre 13).

```
?DONNE "DEPLACEMENTS INTERPRETE REDUIS DEPLACE
```

L'ordinateur attend vos instructions et le contenu de DEPLACEMENTS est créé dès que vous tapez "F. Les actions réalisant votre dessins sont alors dans :DEPLACEMENTS et si vous "videz l'écran" vous retrouverez votre dessin en demandant :

```
?EXEC :DEPLACEMENTS
```

ou les instructions données en demandant :

```
?ECRIS :DEPLACEMENTS
```

Projet 3 : ... Et la maison ! devient un paysage

Dans ce projet, nous allons utiliser le crayon optique pour disposer les éléments d'un paysage sur l'écran.

Tout d'abord les composantes de ce paysage :
un ARBRE
un TRAIN
la MAISON
un BATEAU

Nous utiliserons les 2 dernières lignes de l'écran pour écrire ces mots qui seront pointés par le crayon optique. Par un deuxième pointage, le crayon optique choisira l'emplacement sur l'écran du dessin correspondant. Il sera ainsi possible de créer différents paysages, sur l'écran en indiquant simplement l'emplacement de chaque dessin.

Tout d'abord voyons les procédures réalisant les quatres dessins :

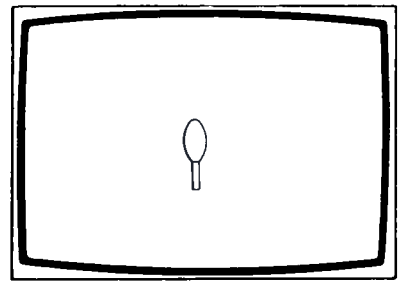
1. Les procédures "utilitaires"

```
POUR TRIANGLE :COTE
REPETE 3 [AVANCE :COTE TD 120]
FIN

POUR RECTANGLE :L1 :L2
REPETE 2 [AVANCE :L1 TD 90 AVANCE :L2 TD 90]
FIN
```

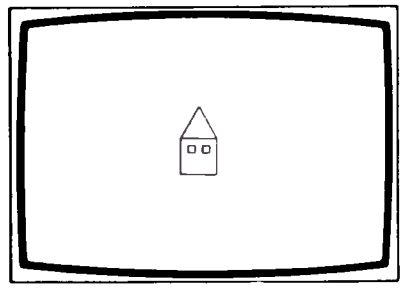
2. Les dessins

a) L'arbre



```
POUR ARBRE
CT BC FCC 3 RECTANGLE 20 5
AVANCE 20 TG 45 FCC 2
REPETE 90 [AV .5 TD 1] TD 45 REPETE 10 [AV .5 TD 1]
TD 35 REPETE 90 [AV .5 TD 1]
FIN
```

b) La maison



```
POUR MAISON
CT BC FCC 7 RECTANGLE 25 25
AVANCE 25 TD 30 FCC 1 TRIANGLE 25
TD 60 LC AV 10 TD 90 AV 5 BC FCC 7 RECTANGLE 5 5
LC TG 90 AVANCE 5 BC
FCC 7 RECTANGLE 5 5
FIN
```

c) Le train se décompose en :

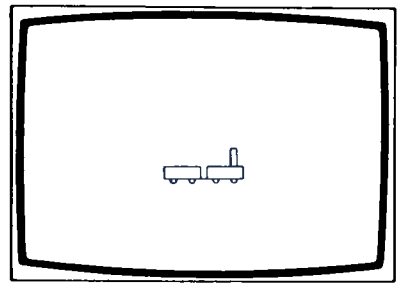
un wagon

```
POUR WAGON
RECTANGLE 15 40 TD 90
AVANCE 5 REPETE 20 [AV 1 TD 18]
AVANCE 30 REPETE 20 [AV 1 TD 18]
FIN
```

une locomotive

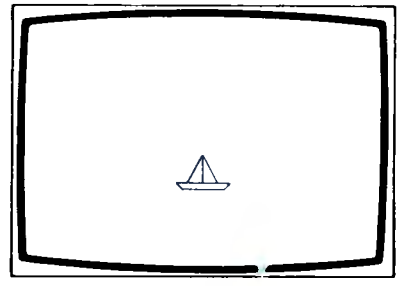
```
POUR LOCO
WAGON
TG 90 LC AVANCE 15 TG 90 AVANCE 5 TD !
90 BC RECTANGLE 15 5
FIN
```

et le train



```
POUR TRAIN
CT BC FCC 5 WAGON AVANCE 15
TG 90 FCC 8 LOCO
FIN
```

d) Le bateau



```
POUR BATEAU
CT BC FCC 8 TG 90 AVANCE 25 TD 45 AVANCE 10
TD 135 AVANCE 60 TD 135
AVANCE 10 TD 45 AVANCE 20
TD 90 LC AVANCE 10 BC FCC 7 AVANCE 20 RECULE 20 TG 90 LC
RECULE 10 BC
FCC 7 TRIANGLE 25
FIN
```

3. Les procédures de mise en place du paysage

La procédure DEPART mettra en place le "menu" c'est-à-dire les différents choix possibles. Il faut bien repérer les zones sur lesquels le nom des quatre dessins sera écrit :

Les mots sont écrits sur la 21e et la 23e ligne de l'écran. Ces lignes correspondent à la position graphique -67 et -75 pour l'axe des Y. Ce repérage nous servira pour le pointage du crayon optique.

ARBRE BATEAU

MAISON TRAIN

Dès la présentation faite, avec le crayon optique il sera possible de faire un CHOIX de dessin puis de le PLACER.

```
POUR DEPART
ME 6 CT FEN FCB 5 FCFT 4
REPETE 8 [EC "]
FCURS [5 21] EC "ARBRE
FCURS [25 21] ECRIS "BATEAU
FCURS [5 23] ECRIS "MAISON
FCURS [25 23] ECRIS "TRAIN
PLACER CHOIX
FIN
```

CHOIX est alors la procédure qui REND à PLACER le nom (mis dans une liste d'un dessin), un BEEP indiquera que le nom du dessin a été pris en compte. Elle permet aussi de sortir du jeu en appuyant sur une touche... par exemple pour faire une COPIE de l'écran :

```
POUR CHOIX
SI TOUCHE? [RENDS LISCAR]
SI CONTACT? [DONNE "POS POSOPT] [RENDS !
CHOIX STOP]
SI VIDE? :POS [RENDS CHOIX]
SI DER :POS < -75 [SI PREM :POS < 0 [BE!
EP RENDS [MAISON]] [BEEP RENDS [TRAIN]]!
]
SI DER :POS < -67 [SI PREM :POS < 0 [BE!
EP RENDS [ARBRE]] [BEEP RENDS [BATEAU]]!
]
RENDS CHOIX
FIN
```

```
POUR BEEP
DUREE 12 JOUE "DO
FIN
```

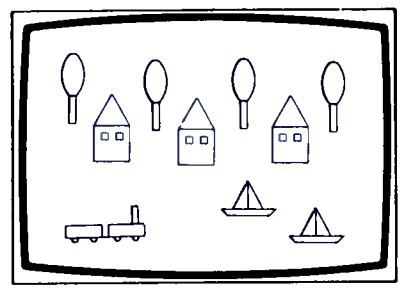
PLACER ce CHOIX consiste alors à sortir de ces procédures si une touche a été frappée, sinon repérer la POSITION du prochain contact du crayon optique afin d'EXECuter le dessin choisi. Ceci est rendu possible par la primitive EXEC qui permet l'exécution du contenu d'une liste. Lorsque le dessin a été exécuté un nouveau CHOIX est permis.

```
POUR PLACER :LISTE
SI MOT? :LISTE [STOP]
SI CONTACT? [LC FPOS POSITION FCAP Ø EX!
EC :LISTE PLACER CHOIX] [PLACER :LISTE]
FIN
```

Enfin POSITION qui REND à PLACER un point de l'écran hors des lignes du "menu" :

```
POUR POSITION
SI CONTACT? [DONNE "POS POSOPT] [RENDS !
POSITION]
SI VIDE? :POS [RENDS POSITION]
SI DER :POS < -6Ø [RENDS POSITION]
RENDS :POS
FIN
```

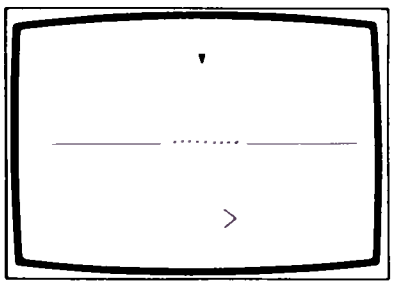
Demandez :
?DEPART
Et faite vos choix!!



Projet 4 : la tortue et l'oiseau

1. Présentation du jeu

Un oiseau est prisonnier. Il voit le ciel à travers des barreaux. Il suffit de casser 3 barreaux contigus pour que l'oiseau s'envole... Le joueur dispose d'un "matériel de tir" et de 1Ø coups pour libérer l'oiseau.



2. Analyse graphique de ce jeu

La présentation graphique du jeu peut se faire sous la forme suivante :

L'oiseau est en bas de l'écran son passage vers le haut est interdit par un mur (trait plein) et une fenêtre avec des barreaux symbolisé par des points.

Le "matériel de tir" sera symbolisé par la tortue.
Elle est située en haut de l'écran et peut se déplacer horizontalement (déplacement commandé par la manette 1) ou se propulser vers le mur en appuyant sur le bouton de la manette 1. Si elle vise soit le mur soit entre 2 barreaux elle ne peut pas passer et revient en arrière. Si elle vise exactement un barreau, le barreau se détache sous sa poussée et une brêche est faite. Dès que 3 barreaux consécutifs ont été enlevés l'oiseau s'envole par le trou.

a) La tortue

Elle se trouve en haut de l'écran et se déplace horizontalement c'est-à-dire que le premier élément de sa POSition varie entre – 16Ø et 159. Lorsque l'instruction lui est donnée de "tirer", si elle vise un barreau, il va s'effacer sinon il ne se passe rien, c'est un "coup" perdu. Dès que 3 barreaux consécutifs ont été effacés, le joueur a gagné l'oiseau est libéré.

Il faut :
1. Repérer exactement l'emplacement des barreaux.
2. Connaître l'état de chacun des barreaux (en place fermé ou effacé ouvert) afin de tester après chaque tir si 3 barreaux consécutifs ont été enlevés.

b) Les barreaux

Décidons qu'il y aura 9 barreaux situés à l'horizontale passant par l'origine, symbolisés par des POINTS de coordonnées [-40 0] [-30 0] [-20 0] [-10 0] [0 0] [10 0] [20 0] [30 0] [40 0]. Il suffit de mettre les abscisses de ces points dans une liste et de tester si la coordonnée horizontale de la tortue appartient à cette liste pour savoir si le tir a été efficace ou non.

Par ailleurs rangeons dans un mot de 9 lettres l'état des barreaux, une lettre correspondant à l'état d'un barreau. Au départ "ETAT sera FFFFFFFFF (tous les barreaux fermés) et dès qu'un barreau aura été touché, le F correspondant se changera en O. Dès que le mot aura trois O consécutif le joueur aura gagné.

c) L'oiseau

Si 3 barreaux côte à côte ont été effacés l'oiseau peut s'envoler. Pour repérer l'emplacement du "trou" il suffira de connaître la place des O consécutifs dans le mot :ETAT...

Tout est en place pour écrire les procédures !

3. Les Procédures

Il nous faut écrire plusieurs procédures :

DEPART réalise le dessin support du jeu.

DEPLACE commande le déplacement, tire si le bouton 1 est appuyé et comptabilise le nombre de tirs.
DEPLACE appelle ANA qui analyse le statut de la manette, et déplace la tortue en conséquence.
TIRER ralentit le tir de la tortue.
Dans TIR, si PREMier POS est dans la liste [-40 -30 -20 -10 0 10 20 30 40]

TIR fait avancer la tortue en effaçant et remplace F par O à la place correspondant au barreau enlevé dans le mot :ETAT. A trois O consécutifs le joueur a gagné, sinon TIR fait avancer la tortue en LC.

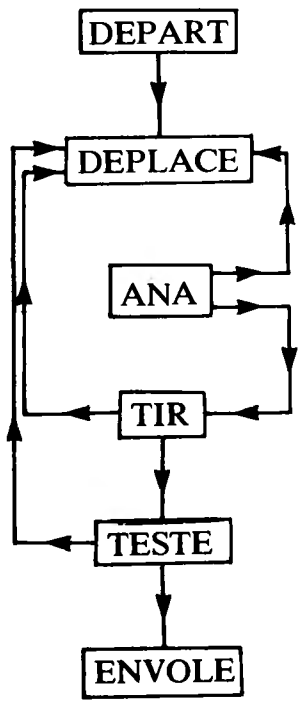
PLACE :MOT :LISTE donne la place d'un élément (ici un mot) dans une liste.

REMPLACE :I :MOT :LETTRE remplace la Ième lettre de :MOT par :LETTRE.

TESTE :MOT rend la place du premier groupe de trois O dans ce mot ou bien le nombre de lettres du mot.

ENVOLE :COORD permet à l'oiseau de s'envoler à la verticale du trou centré sur :COORD.

Les procédures s'organisent de la façon suivante :



mise en place de la scène et
des noms ETAT et BARREAUX

Analyse la position de MANETTE 1
et déplace la tortue en conséquence

Si BOUTON 1 a été appuyé
sinon DEPLACE

fait modifier éventuellement ETAT

Si 3 BARREAUX adjacents sont enlevés

Écrivons d'abord les procédures générales que sont PLACER, REMPLACE et TESTE:

```
POUR PLACER :MOT :LISTE
SI EGAL? PREM :LISTE :MOT [RENDS 1] [R!
ENDS 1 + PLACER :MOT SP :LISTE]
FIN
```

Donne la place du :MOT dans la :LISTE

Remarquons que PLACER n'a de sens que si le contenu de :MOT est un élément du contenu de :LISTE. Ceci sera testé dans la procédure TIR.

```
POUR REMPLACE :I :MOT :LETTRE
SI EGAL? :I 1 [RENDS MOT :LETTRE SP :M!
OT] [RENDS MOT PREM :MOT REMPLACE :I -!
 1 SP :MOT :LETTRE]
FIN
```

Remplace la Ième lettre du :MOT par :LETTRE

```
POUR TESTE :MOT
SI VIDE? :MOT [RENDS 0]
SI EGAL? PREM :MOT "O [SI DEUXAUTRES S!
P :MOT [RENDS 1] [RENDS 1 + TESTE SP :!
MOT]] [RENDS 1 + TESTE SP :MOT]
FIN

POUR DEUXAUTRES :MOT
SI OU VIDE? :MOT VIDE? SP :MOT [RENDS !
VRAI]
SI ET EGAL? PREM :MOT "O EGAL? PREM SP!
 :MOT "O [RENDS VRAI] [RENDS FAUX]
FIN
```

TESTE avec DEUXAUTRES vérifient si MOT contient trois O consécutifs. Si c'est le cas le résultat du test est la place du premier O sinon c'est 9. Par exemple TESTE "FFFOOOFF est 5 alors que TESTE "FOFOFOOFF est 9.

Écrivons les procédures de mise en scène :
La procédure de DEPART qui décrit la scène et appelle le DEPLACEment de la tortue avec en référence l'état initial des barreaux et le numéro de l'essai :

```
POUR DEPART
ENR CT ME 3 FCB 4 FCFG 5 FCFT 5 REPETE!
5 [ECRIS "]
LC FPOS [-160 0] TD 90
BC FCC 0 AVANCE 115 LC
POINTS -40 FPOS [55 0] BC AV 155
DONNE "BARREAUX [-40 -30 -20 -10 0 10 !
20 30 40]
```

```
DONNE "ETAT "FFFFFFFFF
LC FPOS [-50 -50] BC FCC 4 OISEAU
LC FPOS [-20 100] FCAP 180 MT FCC 4
DEPLACE :ETAT 1
FIN
```

Pour compléter la scène il faut écrire la procédure qui dessine l'oiseau et celle qui dessine les barreaux :

```
POUR OISEAU
CT TG 45 AV 10 RE 10
TD 90 AV 10 RE 10 TG 45
FIN

POUR POINTS :X
SI EGAL? :X 50 [STOP]
POINT PH :X 0
POINTS :X + 10
FIN
```

DEPLACE affichera le numéro de l'essai. Elle permet de relancer le jeu lorsque le joueur a perdu, ANAlyse l'état de la manette 1 qui translate la tortue plus ou moins vite, à droite ou à gauche et TIR dès que le bouton de la manette 1 est pressé :

```
POUR DEPLACE :MOT :N
SI EGAL? :N 11 [ECRIS [PERDU! VOULEZ-V!
OUS RECOMMENCER? ( O/N )] SI EGAL? LIS!
CAR "O [DEPART STOP] [STOP]]
ANA MANETTE 1 :MOT :N
FIN

POUR ANA :MAN1 :M :N
SI BOUTON? 1 [TIR :MOT :N STOP]
SI :MAN1 = 0 [DEPLACE :MOT :N STOP]
SI :MAN1 < 6 [LC TG 90 AVANCE :MAN1 TD!
 90] [LC TG 90 RECULE :MAN1 TD 90]
DEPLACE :MOT :N
FIN
```

TIR commande de TIRER en levant le crayon si la tortue ne vise pas le barreau, de gommer si la tortue est face à un barreau puis de TESTER le nouvel état des barreaux :

```
POUR TIR :MOT :N
SI NON MEMBRE? PREM POS :BARREAUX [LC !
TIRER DEPLACE :MOT :N + 1] [FCC - (CF !
+ 1 ) BC TIRER TESTER REMPLACE PLACER !
PREM POS :BARREAUX :MOT "O :N]
FIN

POUR TIRER
REPETE 10 [REPETE 10 [AVANCE 1]]
REPETE 10 [REPETE 10 [RECULE 1]]
FIN
```

TESTER commande l'ENVOLE (face au second O) si le résultat du TESTE n'est pas 9 sinon redonne la possibilité de DEPLACER encore la tortue mais avec le numéro suivant pour l'essai.

```
POUR TESTER :M :N
SI TESTE :M < 8 [ENVOLE ITEM 1 + TESTE!
 :M :BARREAUX] [DEPLACE :M :N + 1]
FIN
```

Enfin l'ENVOLE de l'oiseau face à la brèche :

```
POUR VOLE :X
REPETE :X [BC FCC 4 OISEAU FCC - (CF +!
 1) OISEAU RE 5]
FIN


POUR ENVOLE :COORD
LC FPOS [-50 -50] FCAP 90 BC FCC - (CF!
 + 1) OISEAU
LC FPOS PH :COORD 0 FCAP 180
FCC 4 VOLE 20
FIN
```

Commandez :
?DEPART
et... bonne chance !...

Deuxième partie

Chapitre 13

Espace de travail et sauvegarde des productions

On appelle **espace de travail** tout ce qui se trouve dans la mémoire centrale de votre ordinateur à un moment donné. Il comporte les primitives, les procédures et les noms que vous avez définis.

Cette mémoire fonctionne tant que l'appareil est allumé. Quand on l'éteint, toutes les "connaissances" de la machine se perdent. Cependant, il est possible d'enregistrer votre travail sur une disquette ou une bande magnétique ou encore de l'imprimer.

La sauvegarde sur disquette ou bande magnétique vous permet de ramener dans l'espace de travail ce que vous aviez précédemment fait.

1. L'espace de travail

Certaines primitives permettent de savoir ce qui se trouve dans l'espace de travail à un moment donné et d'autres de le modifier.

a) Afficher

CONTENU rend tout le contenu de l'espace de travail : primitives, noms de choses, noms de procédures et tout ce que vous avez tapé au clavier.
IMNS affiche les noms créés avec les choses qu'ils désignent.
IMTS affiche les titres des procédures.
IM affiche la définition d'une procédure. (IM "CARRE)
IMTOUT affiche tout ce que vous avez créé (noms de choses et procédures).

b) Effacer

EFN efface de l'espace de travail un nom. (EFN "FORMES)

EFP efface de l'espace de travail une procédure. (EFP "CARRE)
.EFT efface de l'espace de travail tout ce que vous avez crée. Tout se passe comme si vous veniez de charger le langage Logo

2. Sauvegarder votre travail

Il est toujours utile de sauvegarder sur cassette ou disquette ce que vous faites en Logo afin de pouvoir le retrouver. Par la suite, vous pouvez améliorer vos productions, les modifier, les utiliser dans d'autres constructions ou simplement les garder en tant que traces du travail effectué.
Si vous voulez enregistrer et relire vos programmes sur cassette, vous devez utiliser le lecteur enregistreur de programmes (LEP).
Celui-ci doit être branché, et connecté à l'ordinateur (grâce au cordon qui porte une prise DIN) *avant d'allumer* l'unité centrale.
Utilisez des cassettes prévues pour les données (bandes certifiées ordinateur) et de courte durée. A défaut, vous pouvez prendre des cassettes normales au Fer (C3Ø ou C45).
Pour enregistrer et relire vos programmes sur disquette, vous aurez besoin d'un ou plusieurs lecteurs de disquettes, d'un contrôleur et d'une disquette système DOS-LOGO. Connectez les lecteurs à l'ordinateur et allumez-les (interrupteur à l'arrière du lecteur) *avant d'allumer* l'unité centrale.
Introduisez la disquette DOS-LOGO dans le lecteur du bas avant de sélectionner le menu proposé à l'écran. Attendez que la lumière du lecteur s'éteigne pour retirer la disquette.
Attention ! n'éteignez pas les lecteurs de disquettes avant d'éteindre l'ordinateur.
Si vous possédez un MO5, après avoir chargé Logo, mettez la disquette marquée " DOS-LOGO " dans le lecteur du bas. Ensuite Tapez : .EFT (Attention ! .EFT réinitialise Logo c'est-à-dire que tout ce qui est en mémoire sera perdu).

a) Enregistrer sur cassette

Introduisez votre cassette dans le LEP (1)

---

(1) Si les lecteurs de disquettes sont aussi connectés à l'appareil, alors commencez par taper :
?FLECTEUR 7
FLECTEUR permet d'établir un lecteur actif ; 7 correspond au lecteur de cassettes. Ceci indique à la machine que vous utiliserez le LEP ; nous disons qu'il est l'unité active.

Rembobinez la bande jusqu'à la bande amorce et enfoncez les touches [•] et [►] du LEP (pour enregistrer).
Vous pouvez maintenant enregistrer sur cassette ce que vous voulez garder. Il faut donner un nom à l'ensemble que vous voulez enregistrer : c'est *le nom du fichier*.

Ainsi, après le Chapitre 1 vous pouvez, par exemple, garder vos procédures en tapant :

```
?SAUVE "CHAP1 [BONJOUR UNTRAJET ESCALIER]
```

SAUVE est la primitive qui permet d'enregistrer sur cassette ou disquette.
CHAP1 est le nom du fichier (précédé de guillemet). C'est un mot ayant au plus 8 caractères.
La liste des procédures à sauvegarder (une ou plusieurs) est écrite entre crochets [BONJOUR UNTRAJET ESCALIER].
Faites attention de sauvegarder aussi ce dont une procédure dépend.

Prenons un autre exemple. Dans le Chapitre 6, nous avons donné une adresse à la maison en faisant DONNE "MAISON [2 RUE DE LA TORTUE]. La procédure ADRESSE trace une MAISON de 60 pas et écrit :MAISON, c'est-à-dire 2 RUE DE LA TORTUE. D'autre part, la procédure MAISON dépend des procédures TRIANGLE et CARRE (voir Chapitre 4). Comment sauvegarder tout ça ?

Nous pouvons regrouper les procédures CARRE et TRIANGLE sous un nom

```
?DONNE "FORMES [CARRE TRIANGLE]
```

et ensuite

```
?SAUVE "RUE [ADRESSE MAISON "MAISON :FORMES]
```

Examinons cette liste :
Seront sauvegardées les procédures ADRESSE et MAISON, le nom "MAISON avec sa chose (DONNE "MAISON [2 RUE DE LA TORTUE]) et les procédures CARRE et TRIANGLE contenues dans FORMES (1).

Après avoir appuyé sur [ENTREE] le curseur disparaît et la machine exécute votre commande. Il faut attendre l'apparition du symbole d'invite ? et du curseur pour que l'enregistrement soit terminé.

---

(1) Si on mettait un guillemet au lieu de deux points devant FORMES, on sauvegarderait le nom FORMES avec la chose qu'il désigne c'est-à-dire la liste [CARRE TRIANGLE] et non pas les procédures correspondantes.

Vous pouvez alors continuer votre travail ou éteindre vos appareils.

Attention ! pensez à noter sur un papier le nom du fichier et le numéro à partir duquel il commence et celui auquel il se termine. Vous releverez ce numéro sur le compteur du LEP. Ceci permet de repérer le début du prochain enregistrement et vous aurez des indications pour le ramener en mémoire centrale.

b) Enregistrer sur disquette

Pour pouvoir enregistrer votre travail sur disquette, il faut d'abord l'initialiser, la formater.

Attention ! Prenez une disquette vierge ou une disquette dont le contenu peut être détruit. Si vous initialisez une disquette ayant déjà un contenu, tout sera effacé et les données qu'elle contient seront perdues.

Introduisez votre disquette (l'étiquette vers l'extérieur et vers le haut) dans le lecteur de disquettes. Fermez le lecteur en descendant le levier. Tapez :

```
?FORMATE 1
```

FORMATE est la primitive permettant d'**initialiser** une disquette. Sa donnée, Ø, 1, 2 ou 3 indique le numéro du lecteur utilisé (lecteur 80 Ko) ou le numéro de la face (lecteur 320 Ko).

Lecteur 80 Ko

Ø	1er lecteur
1	2e lecteur
2	3e lecteur
3	4e lecteur

Lecteur 320 Ko

	face	lecteur
Ø	1re	1er
1	2e	
2	1re	2e
3	2e	

Le formatage est compatible avec celui du BASIC ; vous pouvez donc aussi bien enregistrer des fichiers Logo que des fichiers Basic sur la même disquette.

Il faut attendre que le ? et le curseur réapparaissent pour faire autre chose. Ceci prend quelques minutes.

La disquette formatée contient 40 pistes de 16 secteurs. Après initialisation il n'y a rien d'inscrit au catalogue de la disquette.

Pour enregistrer, introduisez une disquette initialisée dans le lecteur de disquettes. Le lecteur actif par défaut, est le lecteur 0 (celui du bas) (lecteur 80 Ko) ou la face 0 (lecteur 320 Ko).

Si vous voulez changer de lecteur, utilisez la primitive FLECTEUR. Ensuite, comme pour sauvegarder sur cassette, il faut taper par exemple

```
?SAUVE "RUE [ADRESSE MAISON "MAISON :FORMES]
```
(1)

et attendre que le curseur réapparaisse sur l'écran pour faire autre chose.

3. Ramener votre travail

Voyons maintenant comment retrouver dans votre espace de travail un fichier déjà enregistré.

a) Cassette

Il n'y a pas de primitive permettant de voir tous les noms des fichiers enregistrés sur cassette (contrairement aux disquettes).

Pour savoir quel fichier vous voulez ramener, vous devez donc consulter le document sur lequel vous avez noté leurs noms.

Pour ramener un fichier enregistré sur cassettes :
a) Rembobinez la bande jusqu'au numéro du compteur du début du fichier.
b) Enfoncez la touche [►] du LEP
c) Tapez :

```
?RAMENE "RUE
```
par exemple

Tous les noms de procédures contenues dans ce fichier apparaissent sur votre écran comme si vous veniez de les définir.

Attendez l'arrêt du LEP avant de faire autre chose.

---

(1) Si vous donnez un nom de fichier qui existe déjà sur la disquette vous aurez le message :

```
EXISTE DEJA
```

b) Disquette

Introduisez votre disquette dans le lecteur actif, et tapez :

```
?CATALOGUE
```

Peu après, apparaissent sur l'écran tous les noms des fichiers enregistrés sur cette disquette suivis du nom du langage dans lequel ils ont été écrits (LOG pour Logo)

Chosissez celui dont le contenu vous intéresse et tapez, par exemple :

```
?RAMENE "RUE
```

Tous les noms de procédures contenues dans ce fichier apparaissent sur votre écran comme si vous veniez de les définir. Vous pouvez les utiliser comme vous voulez.

4. Faire le ménage dans vos fichiers

a) Sur cassette

Pour effacer un fichier de la cassette vous pouvez, comme sur un magnétophone ordinaire, enregistrer autre chose par dessus.

b) Sur disquette

- Éliminer un fichier

Si vous tapez :

```
?DETRUIS "RUE
```

Ce fichier et son contenu disparaîtront de la disquette.
Néanmoins, ils feront encore partie de l'espace de travail en cours si vous l'aviez ramené.

- Éliminer certaines procédures d'un fichier

Ramenez le contenu du fichier dans l'espace de travail et supprimez ce fichier de la disquette (DETRUIS "nom).

Ensuite, sauvez les procédures qui vous intéressent dans un nouveau fichier portant le même nom ou un autre.

- Combiner-séparer deux fichiers

Si vous voulez combiner deux fichiers, ramenez-les successivement dans l'espace de travail. Ensuite supprimez les noms de ces deux fichiers de la disquette et sauvez toutes les procédures de votre espace de travail sous le nom d'un nouveau fichier.

De même, vous pouvez séparer un fichier en plusieurs. Ramenez-le dans l'espace de travail, supprimez-le de la disquette et sauvez les procédures

dans des fichiers distincts. Une procédure peut évidemment faire partie de plusieurs fichiers si son nom figure dans chacun d'eux.

5. L'imprimante

Votre ordinateur peut commander une imprimante. Vous pouvez donc garder votre travail en le faisant imprimer sur papier. **Attention !** l'imprimante doit être connectée et sous tension avant d'allumer l'ordinateur.
Quand vous voulez l'utiliser, tapez :

```
?SORTIE 2          (1)
```

Tout ce qui s'affichera à l'écran sera alors imprimé. Si vous tapez IMTOUT, tout votre travail sera imprimé.

Pour arrêter l'imprimante :

— appuyez simultanément sur les touches [CNT] [C]

— ou tapez :

```
?SORTIE 1          (2)
```

lorsque vous avez terminé.

Après l'avoir mise en marche, vous pouvez utiliser l'imprimante de différentes façons :

• Imprimer un travail terminé

Il est possible d'imprimer une procédure déjà définie (en demandant par exemple, IM "MAISON, ou l'ensemble de votre espace de travail, (IMTOUT), ou simplement les titres de vos procédures, (IMTS).

• Imprimer en temps réel

Vous pouvez aussi imprimer votre travail en même temps que vous le faites. L'imprimante mise en route, chaque fois que vous appuyez sur la touche [ENTREE] la ligne terminée sera en même temps imprimée et exécutée.
Si vous possédez une imprimante thermique, la primitive COPIE imprime sur papier l'écran tel que vous le voyez : avec les dessins.

---

(1) 2 est le code pour l'imprimante parallèle.

(2) SORTIE 1 pour écrire seulement à l'écran.

Chapitre 14

Le monde de l'éditeur

L'éditeur est celui qui fait paraître un texte, le présente, l'annote etc.

En Logo, l'éditeur est un outil mis à votre disposition pour faciliter votre travail. On le considère comme un monde à part.

Dans ce monde de l'éditeur, le curseur peut se déplacer à n'importe quel endroit de l'écran. Il est possible ainsi, de modifier, insérer ou supprimer des caractères, voire des lignes et des paragraphes entiers sans la moindre "rature" !

Vous pouvez utiliser l'éditeur, non seulement pour vos textes Logo mais aussi comme support d'écriture pour composer des textes en français.

L'éditeur est un "univers de planification". Aucune instruction écrite ne saurait être exécutée dans l'immédiat.

1. Entrer — sortir de l'éditeur

a) Entrer

Pour accéder à l'éditeur il faut taper :

?ED [ENTREE]

L'écran devient bleu foncé et le bord bleu clair. Ces repères rappellent qu'on se trouve à l'intérieur du monde particulier de l'éditeur. Le symbole d'invite, ?, n'est plus là. Seul le curseur apparaît.

Vous pouvez maintenant écrire un texte en français, des instructions en Logo, définir une procédure ou des noms.

b) Sortir

Pour sortir de l'éditeur et renouer avec le mode direct, vous avez deux possibilités :

1. [CNT] [Q] En appuyant simultanément sur les touches [CNT] et [Q], l'édition est conclue, vous quittez l'éditeur et ce que vous avez écrit entre dans l'espace de travail. Les noms des procédures définies s'affichent, si le titre d'une procédure est mal écrit, un message Logo apparaîtra.

2. CNT C Vous arrêtez le travail de l'éditeur et renouez avec le mode direct. Ce que vous avez écrit, n'entre pas dans votre espace de travail ; tout reste cependant dans une **mémoire à part,** propre à l'éditeur. Vous pouvez alors sauver le contenu de cette mémoire sur cassette ou disquette et le recharger plus tard.

2. Manipulations du curseur dans le monde de l'éditeur

Les 4 flèches sur la droite du clavier permettent de déplacer le curseur sur l'écran :

← vers la gauche. Si le curseur est en début de ligne, il remonte à la fin de la ligne précédente (si elle existe).
→ vers la droite. Si le curseur est en fin de ligne, il se rend au début de la ligne suivante (si elle existe).
↓ vers le bas, même colonne.
↑ vers le haut, même colonne.

La touche EFF permet d'effacer le caractère situé à la place du curseur.

La touche ENTREE insère une ligne vide à la position du curseur. En appuyant sur cette touche, c'est comme si vous tapiez un caractère qui peut par conséquent être supprimé ou inséré dans un texte. Par exemple, à la fin d'une ligne si vous appuyez sur la touche EFF la ligne inférieure remonte à la position du curseur.

D'autres combinaisons de touches sont utiles :

CNT D supprime le caractère qui se trouve immédiatement à gauche du curseur (et qui peut être un espace ou un retour à la ligne).
CNT S supprime le reste de la ligne à droite du curseur
CNT R permet de rappeler la dernière suppression effectuée par CNT S et de l'insérer à la position du curseur.

Lorsqu'on utilise la touche CNT il faut y maintenir la pression dessus pendant qu'on appuie sur une autre touche.

Et pour vous déplacer dans le texte :
INS ← positionne au début de la ligne courante (1)
INS → positionne à la fin de la ligne courante

(1) On appelle **ligne courante,** la ligne sur laquelle se trouve le curseur.

INS ↑ positionne au début du texte
INS ↓ positionne à la fin du texte
INS INS annule une pression de INS non désirée.

Dès que vous tapez sur INS le bord du tableau devient vert : l'éditeur attend une instruction. Ensuite appuyez sur la touche voulue, le bord du tableau redevient bleu clair.

Quand vous vous trouvez proche de la saturation de l'éditeur, le bord de l'écran clignote rouge. Si vous avez atteint la fin de la place disponible, que ce que vous tapez est refusé, CNT Q ou CNT C permettent de tout arrêter avec leurs fonctions respectives.

3. Éditeur et procédures

L'éditeur est un instrument précieux quand on définit des procédures en Logo. Il permet de les modifier ou d'en créer de nouvelles.

a) Définir une procédure

Vous pouvez définir une nouvelle procédure en entrant directement dans l'éditeur (ED). Comme d'habitude, la première ligne contient POUR avec le nom de la procédure et ses données, ensuite les lignes d'instructions, et la dernière ligne contient uniquement le mot FIN.

Après avoir défini une procédure, vous sortez de l'éditeur en appuyant simultanément sur CNT Q : ainsi vous renouez avec le mode direct et le message habituel apparaît :

VOUS VENEZ DE DEFINIR ...

Vous pouvez maintenant demander l'exécution de cette nouvelle procédure qui vient d'être définie.

Plusieurs procédures

Si vous voulez définir plusieurs procédures à la suite, n'oubliez pas d'écrire le mot FIN avant d'entreprendre la définition suivante qui doit commencer également par POUR.

Corriger une erreur

Pour corriger une erreur dans l'éditeur la démarche est simple :

— amenez le curseur à la place du caractère à modifier en le déplaçant par les touches décrites précédemment

— effacez le ou les caractères à modifier, tapez les nouveaux, et ensuite ramenez le curseur à la bonne position pour continuer la définition.

Si votre définition est terminée, vous n'avez pas besoin de ramener le curseur à un endroit précis avant de faire CNT Q.
Dans l'éditeur, une ligne Logo (conclue par la touche ENTREE) peut contenir plus de 248 caractères, à condition qu'il n'y ait pas de mot (suite de caractères) de plus de 248 caractères.

A la fin d'une ligne écran, un point d'exclamation (!) indique que la ligne Logo continue.

b) Corriger une procédure définie en dehors de l'éditeur

Vous pouvez également vous servir de l'éditeur pour corriger des procédures définies à l'extérieur de celui-ci.

Erreur découverte lors de l'écriture

Si pendant l'écriture d'une procédure en dehors de l'éditeur vous vous apercevez d'une faute que vous ne pouvez pas corriger directement, vous avez deux solutions :

a) Taper sur une nouvelle ligne le mot FIN comme si la procédure en question était terminée. Ceci vous permet de revenir au mode direct. Ensuite vous amenez votre procédure dans l'éditeur (ce qu'on appelle "éditer une procédure") en tapant par exemple :

```
?ED [MAISON]    ENTREE
```

(mettez le ou les noms de procédures entre crochets)

Vous vous retrouvez dans l'éditeur avec la procédure MAISON comme vous venez de l'écrire. Vous pouvez maintenant manipuler le curseur pour la modifier.
Après avoir porté vos modifications, pour pouvoir continuer sa définition :

— ramenez le curseur sur la fin de l'avant dernière ligne (avant le mot FIN)
— appuyez sur ENTREE : une nouvelle ligne vide est créée
— vous pouvez continuer à écrire vos instructions

Quand la définition est terminée, il n'y a pas besoin de réécrire le mot FIN. Appuyez sur CNT Q pour quitter l'éditeur en validant la procédure.

b) Continuer la définition jusqu'à la fin, en ignorant l'erreur. Ensuite, au lieu de passer tout de suite à l'exécution de la procédure, éditez-la

(ED [MAISON] par exemple). Vous pouvez ainsi la corriger dans l'éditeur.

Appuyez sur CNT Q pour quitter en validant vos corrections.

Erreur découverte lors de l'exécution

Quand vous demandez l'exécution d'une procédure, le résultat ne correspond parfois pas à celui attendu. Il se peut même qu'un message apparaisse indiquant l'impossibilité d'exécuter la procédure.

Il vous faut alors éditer la procédure, bien regarder les instructions données et essayer de comprendre ce qu'elles font. Modifiez ce qui vous paraît être faux en déplaçant le curseur et ensuite validez vos corrections en quittant l'éditeur par CNT Q.

Redemandez l'exécution de la procédure. Il est toujours utile de le faire en sortant de l'éditeur. De cette façon vous voyez si les corrections portées sont les bonnes.

c) Modifier une procédure pour en faire une nouvelle

Vous pouvez créer de nouvelles procédures à partir de celles que vous avez déjà définies.

Pour ceci, éditez la procédure que vous voulez transformer. Après avoir porté les modifications, changez le nom de cette procédure dans l'éditeur. Vous pouvez par exemple ajouter un caractère à la fin du titre.

En sortant de l'éditeur (par CNT Q) vous gardez ainsi dans votre espace de travail deux procédures : l'ancienne et la nouvelle version.

d) Éditer un nom, éditer un groupe de procédures

Si en dehors de l'éditeur vous avez créé un nom désignant une chose, par exemple :

```
?DONNE "PLOUM [UN PETIT CANICHE NOIR]
```

et que vous voulez porter une correction, vous pouvez le ramener dans l'éditeur en tapant :

```
?ED ["PLOUM]
```

Le guillemet devant le mot indique à la machine qu'il s'agit d'un nom désignant une chose qu'il faut éditer et non pas une procédure.
Logo permet aussi de grouper des procédures ou d'autres noms pour les éditer ou les sauver (voir Chapitre 13).

Il s'agit de donner un nom à la série de procédures que vous voulez grouper.
Par exemple :

```
?DONNE "FORMES [CARRE TRIANGLE HEXA]
```

Si vous demandez :

```
?ED [:FORMES]
```

dans l'éditeur apparaîtront les définitions des trois procédures que recouvre FORMES. Ceci est équivalent à :

```
?ED [CARRE TRIANGLE HEXA]
```

Si vous demandez :

```
?ED ["FORMES :FORMES]
```

dans l'éditeur apparaîtront non seulement les définitions des procédures mais aussi le nom FORMES avec sa chose, c'est-à-dire la liste :

[CARRE TRIANGLE HEXA]

Supposons que FORMES soit par ailleurs le nom d'une procédure qui dessine un cercle, alors :

```
?ED [FORMES "FORMES :FORMES]
```

amènera à l'éditeur
— la définition de la procédure FORMES
— le nom FORMES avec sa chose
— la définition des procédures désignées par le nom FORMES.

4. Éditeur et écriture

L'éditeur est un instrument utile pour la composition de textes.

Plus pratique qu'une machine à écrire, ce moyen vous permet d'effacer ou d'ajouter des caractères, voire des paragraphes entiers sans qu'aucune rature n'apparaisse.

Vous pouvez ainsi composer des textes libres, des lettres officielles, ou autres documents. Vous pouvez enfin écrire en mettant des accents, graves, aigus et circonflexes, en utilisant le ç, et tout ce qui fait du bon français !

Quand vous écrivez un texte qui n'est pas du Logo, il est préférable d'appuyer sur les touches CNT C pour sortir de l'éditeur. Ce texte ne sera pas interprété dans votre espace de travail Logo, mais il sera gardé

dans la mémoire tampon propre à l'éditeur dont le contenu peut être supprimé ou sauvé sur cassette ou disquette, ou imprimé.

— Pour écrire en **minuscules :**
Appuyer simultanément sur [•] et la barre d'espacement.

— Pour repasser en **majuscules :**
Appuyer à nouveau simultanément sur [•] et la barre d'espacement.

— Pour **accentuer** une lettre minuscules :
Avant d'écrire la lettre taper [ACC] puis frapper sur l'accent désiré (• et 7 pour l'accent aigu...) et ensuite sur la lettre.

— Pour mettre une cédille (sous un c minuscule) :
Taper [ACC] puis deux fois sur la lettre C.

5. Sauvegarder le contenu de l'éditeur

Après avoir mis votre cassette dans le lecteur enregistreur de programmes ou votre disquette dans le lecteur de disquette (voir Chapitre 13 pour la description des manipulations), tapez par exemple :

```
?SAUVED "FORMES
```

Ainsi, le contenu de l'éditeur, sera sauvé dans un fichier nommé FORMES.

De même, vous pouvez amener dans l'éditeur un fichier se trouvant sur une cassette ou une disquette, sans que ce fichier soit interprété par Logo.

Dans ce cas, après l'avoir positionné dans le lecteur, vous taperez, par exemple :

```
?CHARGE "FORMES
```

Le contenu du fichier apparaîtra dans l'éditeur comme si vous veniez de l'écrire. Demandez :

```
?ED
```

et vous verrez !

6. Imprimer le contenu de l'éditeur sur papier

Si vous voulez imprimer ce que vous avez écrit dans l'éditeur, vous pouvez taper :

```
?ENTREE 4 SORTIE 2 REPETE 1000 [ECRIS LL]
```

Tout ce qui se trouve dans l'éditeur sera imprimé sur papier, que ce soit du Logo ou pas.
Appuyez sur CNT C pour arrêter l'imprimante.

INDEX

A
accents 13
AVANCE , AV 16

B
barre d'espacement 13
BC 29
BOUTON? 114

C
CAP 42, 50
CAR voir Manuel de Référence
cassette 142, 145
CATALOGUE 146
CC 41
CF 43
chaîne de caractères 78
champ 50
CHARGE 154
chose 80
CHOSE 79, 123
clavier 13
CLOS 51
code couleur 50
concaténation 85
CONTACT? 111
CONTENU 141
coordonnées graphiques 50
COPIE 147
crayon de la tortue 50
crayon optique 111
crochets 13, 80
CT 30
curseur 11, 149

D
décimales 85
définition de procédure 18, 21, 22, 150
DER 86, 92
DÉTRUIS 146
disquettes 142, 144, 146
DIV 84
DONNE 79, 80, 81, 82
donnée 54, 61
D.O.S. 142
DURÉE 116

E
écran 24, 52
ECRIS , EC 15
ED 148
éditeur 148
EFN 141
EFP 18, 142
.EFT 142
EGAL? 67
élément 86
ENR 43, 52
ENTREE 12, 15, 21
ENTREE 155
espace 22
espace de travail 141
ET 110
EXEC 127, 128
expression conditionnelle 75, 91

F
FCAP 42, 50
FCC 41, 50

FCFG 43, 52
FCFT 90
FCT 90
FCURS 132
FEN 43, 51
fichier 142
FIN 18, 25
FLECTEUR 142
formatage 144
FORMATE 144
FPOS 42, 51

G
guillemet (") 78, 80

H
HASARD 87

I
IM 141
IMNS 141
imprimante 147, 155
IMTS 141
IMTOUT 141
infixée 91
initialiser 144
instruction 22
ITEM 88

J
JOUE 116

L
LC 29
lecteur-enregistreur de programmes 142
lecteur de disquettes 142
ligne Logo 21, 151
LISCAR 100
liste 79, 80
LISTE? 94
LL 94, 99

M
MANETTE 114
manipulations du curseur 149
MD 100
ME 24
MEMBRE? 90
mémoire centrale 145
mémoire-tampon 154
messages Logo 23
mode direct 17, 21, 22
mode de définition de procédures 18, 21, 22, 150
mot 80
MOT 85
MP 100
MT 16

N
nom de chose 80
nom de fichier 142
nombre négatif 83
NON 110

O
objet Logo 63, 80
OCTAVE 117
opérateurs logiques 109
opérations 91, 107
opérations arithmétiques 64, 91
ORIGINE 52
OU 110

P
parenthèses 108
PH 85
PLP? 106
POINT 138
points (:)
POS 42, 51
POSOPT 111
POUR 18, 25
préfixée 91
PREM 86, 92
primitive 17
procédure 17, 25, 150
PROD 84

Q
QUOT 84

R
RAMENE 145, 146
RECULE , RE 16
récursivité 38, 76, 92, 108
RENDS 102, 108
REPETE 28
RESTE 84

S
SAUVE 143
SAUVED 154
SD 86, 92
SI 75
SOMME 83
SORTIE 147
SP 86, 92
STOP 75
symbole d'invite 11, 21

T
TAPE 90
TD 16
TEMPO 118
test d'arrêt 75
TG 16
TIMBRE 118
tortue 16, 24, 38
troncation 85

V
VE 17
VT 123
VIDE? 86

Liste alphabétique des primitives Logo

A
ASCII
AVANCE , AV

B
BC
BC?
BOUTON?

C
CAP
CAR
CATALOGUE
CC
CF
CHARGE
CHOSE
CLOS
COMPTE
CONTACT?
CONTENU
COPIE
COS
CT

D
DER
DETRUIS
DIFF
DIV
DONNE
DUREE

E
ECH
ECRIS , EC
ED
EFN
EFP
EGAL?
ENR
ENT
ENTREE

ET
EXEC

F
FCAP
FCB
FCC
FCFG
FCFT
FCT
FCURS
FECH
FEN
FIN
FLECTEUR
FLI
FORMATE
FPOS

H
HASARD

I
IM
IMNS
IMTOUT
IMTS
ITEM

J
JOUE

L
LC
LECTEUR
LISCAR
LISTE
LISTE?
LOGO
LL

M
MANETTE
MD
ME
MEMBRE?
MOT
MOT?
MP
MT

N
NETTOIE
NOM?
NOMBRE?
NON

O
OCTAVE
ORIGINE
OU

P
PH
PLACE
PLG?
PLP?
POINT
POS
POSOPT
POUR
PREM
PRIM?
PROC?
PROD

Q
QUOT

R
RAMENE
RC
RECULE, RE
RECYCLE
RENDS
REPETE
RESTE

S
SAUVE
SAUVED
SD
SI
SIN
SOMME
SORTIE
SP
STOP

T
TAPE
TD
TEMPO
TG
TIMBRE
TOUCHE?

V
VE
VIDE?
VISIBLE?
VT

/
*
−
+
<
>
≐

commençant par un point
.CHB
.DEP
.EFT
.EXA
.RES
.ROUT
.SER

Imprimé en France par OFFSET-AUBIN à Poitiers. D.L. juin 1985. Imprimeur nº P 13547